



Universidad
Carlos III de Madrid
www.uc3m.es

Departamento de Teoría de la Señal y de la Comunicación

TRABAJO FIN DE GRADO

Localización de fuentes de emisiones de contaminantes mediante redes de sensores

Autor: Francisco David Reyes Guerrón

Tutor: Francisco Javier González Serrano

Leganés, Junio de 2016

Agradecimientos

Todo camino empieza con un pequeño paso, y como todos llega a su fin.

En esta etapa próxima a terminar, he vivido emociones de todo tipo: tristeza, rabia, ilusión, frustración, pasión, alegría... Lo importante es que ha sido inolvidable y que no me arrepiento ni un solo segundo del camino elegido.

En primer lugar, quiero dar las gracias a Carina, la mujer de mi vida, siempre a mi lado apoyándome, en los buenos momentos y en los no tan buenos. Gracias por todo Marmo.

Papá, mamá, en todo momento incondicionales, gracias por vuestra paciencia y cariño.

A mi familia, gracias por vuestro apoyo.

Gracias también a mi tutor, Francisco Javier González Serrano por su paciencia.

Tabla de contenido

1. ESTADO DEL ARTE	9
1.1 INTRODUCTION	9
1.2 TRABAJOS RELACIONADOS CON EL MISMO PROBLEMA.....	10
2. MODELO DE DISPERSIÓN DE CONTAMINANTES.....	11
2.1 MODELO GAUSSIAN PLUME.....	12
<i>Ventajas y desventajas del modelo</i>	<i>13</i>
<i>Usos</i>	<i>14</i>
3. SENSORES Y MODELO DE MEDIDAS	15
4. SOLUCIONES PROPUESTAS	17
4.1 ALGORITMOS DE OPTIMIZACIÓN	17
4.1.1 <i>Método del Gradiente Descendente.....</i>	<i>18</i>
4.1.2 <i>Método del Gradiente Adaptativo para la localización de fuentes de contaminantes.....</i>	<i>21</i>
4.1.3 <i>Método de Newton para la localización de fuentes de contaminantes</i>	<i>22</i>
4.2 ESCENARIOS DE ESTUDIO	23
4.3 LOCALIZACIÓN DE FUENTES DE CONTAMINANTE BASADA EN EL MODELO DE PROPAGACIÓN	24
4.3.1 <i>Aplicación del método del Gradiente descendente para la localización de fuentes de contaminantes</i>	<i>25</i>
4.3.2 <i>Aplicación del método del Gradiente Adaptativo para la localización de fuentes de contaminantes</i>	<i>32</i>
4.3.3 <i>Aplicación del método de Newton para la localización de fuentes de contaminantes.....</i>	<i>35</i>
4.3.4 <i>Conclusiones del uso de los métodos Gradiente Descendente, Gradiente Adaptativo y método de optimización de Newton</i>	<i>37</i>
4.4 LOCALIZACIÓN DE FUENTES DE CONTAMINANTES INDEPENDIENTEMENTE DEL MODELO DE PROPAGACIÓN	38
4.4.1 <i>Optimización de la función FRG para localizar fuentes de contaminantes.....</i>	<i>39</i>
5. CONCLUSIONS	49

6. LÍNEAS FUTURAS.....	51
7. PRESUPUESTO DE PROYECTO	53
8. BIBLIOGRAFÍA	55
ANEXO 1: FÓRMULAS DE INTERÉS	59
DERIVADAS PARCIALES DEL MODELO GAUSSIAN PLUME:.....	59
ABSTRACT	61
INTRODUCTION	61
GAUSSIAN PLUME MODEL	61
PROPOSED SOLUTIONS	62
CONCLUSIONS.....	67
FUTURE DEVELOPMENTS	68

Tabla de ilustraciones

Ilustración 1: Modelo Gaussian Plume	12
Ilustración 2: Ejemplo de una red de sensores	16
Ilustración 3: Ejemplo de método del Gradiente descendente sobre la función $f(x) = x^2$; $\lambda = 0.1$	18
Ilustración 4: Ejemplo de método del Gradiente descendente sobre la función $f(x) = x^2$; $\lambda = 0.4$	20
Ilustración 5: Ejemplo de método del Gradiente descendente sobre la función $f(x) = x^2$; $\lambda = 0.9$	20
Ilustración 6: Ejemplo de método del Gradiente descendente sobre la función $f(x) = x^2$; $\lambda = 1.1$	21
Ilustración 7: Ejemplo de escenario con 30 sensores.....	24
Ilustración 8: Escenario 1 - Aplicación del Gradiente Descendente (I).....	29
Ilustración 9: Escenario 1 - Aplicación del Gradiente Descendente (II)	29
Ilustración 10: Resultado final de aplicación del Gradiente Descendente.....	30
Ilustración 11: Resultados obtenidos para el escenario 1 cuando se aplican distintos valores de constante de aprendizaje	30
Ilustración 12: Escenario 2 - aplicación del método del Gradiente Descendente.....	31
Ilustración 13: Escenario 3 - Aplicación del Gradiente Descendente.....	32
Ilustración 14: Escenario 3 - Aplicación del método del Gradiente Adaptativo	33
Ilustración 15: Escenario 2 - Aplicación del método del Gradiente Adaptativo	34
Ilustración 16: Evolución del error frente al número de sensores e iteraciones considerado	34
Ilustración 17: Escenario 1 - Aplicación del método de Newton	36
Ilustración 18: Error posición estimada de la fuente frente al número de sensores e iteraciones considerado	37
Ilustración 19: Función FRG (I)	42
Ilustración 20: Función FRG (II)	42
Ilustración 21: Función FRG (III).....	42
Ilustración 22: Optimización de función FRG para la localización de fuentes (II)	44

Ilustración 23: Optimización de función FRG para localización de fuentes (I)	44
Ilustración 24: Optimización de función FRG para localización de fuentes (III).....	44
Ilustración 25: 3 Ejemplos de aplicación del algoritmo de localización utilizando función FRG y aplicando error de offset	45
Ilustración 26: Localización de dos fuentes mediante ajuste de función FRG (I)	47
Ilustración 27: Localización de dos fuentes mediante ajuste de función FRG (II).....	47
Ilustración 28: Localización de dos fuentes mediante ajuste de función FRG (III)	48
Ilustración 29: Localización de dos fuentes mediante ajuste de función FRG (IV)	48

1. Estado del arte

1.1 Introduction

Nowadays, particular attention is being paid to the release of pollutants through several pollution sources. Day after day, the health of any living being is harmed because of the pollution generated by different industrial activities or the burning of fossil fuels, among others.

According to the UN, “Pollution in cities contributes to almost 3.4 million of premature deaths around the world and it is a key factor in respiratory and cardiovascular diseases”, which is a matter of concern.

This Degree Final Project is aimed at identifying the location of these pollution sources in order to carry out a more efficient control over air pollution.

To undertake this task, we must have a sensor network that allows collecting measurements of pollutant concentration in a given context, as well as other interesting parameters, such as wind speed and direction, to process them with a set of iterative algorithms which facilitate the localization of the source.

Lacking a real scenario to perform the relevant tests, it was decided to choose a full simulation by means of the so-called Matlab development environment and the aforementioned Gaussian Plume model. Thus, we can obtain the simulated measurements of pollutants which sensors would collect in a real scenario.

The algorithms will use these measurements to reproduce the pollutant propagation model and therewith locate the source.

1.2 Trabajos relacionados con el mismo problema

Con el paso del tiempo se han desarrollado diversas líneas de investigación, que han intentado dar solución al problema de la localización de fuentes y a la reducción o control de la contaminación. Estas líneas se dividen básicamente en tres partes: problemas de localización de fuentes, problemas de identificación de contaminantes y monitorización de la contaminación.

En algunos casos se ha intentado abordar el problema de la localización de fuentes mediante funciones de probabilidad [22]. En este caso se utiliza una sola estación receptora donde se miden niveles de concentración de contaminantes, velocidad y dirección del viento. De esta forma se relacionan estas medidas entre sí y se consigue averiguar en qué dirección se encuentran varias fuentes de contaminación. Sin embargo, al tener en cuenta un solo receptor no se puede conseguir una localización exacta de la posición de la fuente debido a que no se posee suficiente información.

Es por este motivo, que en este Trabajo de Fin de Grado, se aborda el problema de localización con una red de sensores, porque quizá es más adecuado aprovechar las diferentes medidas que proporcionan los sensores, ya que éstas dan mayor información de la posición de la fuente si se analizan correctamente.

De la misma forma, también se habían realizado con anterioridad proyectos en los que se utilizaba el modelo Gaussian Plume para localizar fuentes, partiendo de la hipótesis de que si el modelo era capaz de generar medidas de concentración de contaminantes, también serviría para abordar el problema inverso [11]. Una de las principales desventajas que se halló fue la necesidad de simplificar el problema asumiendo una serie de hipótesis que derivarían en una menor precisión en la localización para escenarios reales.

Esto se debe a que el problema de la localización de fuentes es muy complicado y ciertas veces es inevitable la simplificación. En gran medida, en este Trabajo de Fin de Grado se intenta mejorar este proceso y aproximarse poco a poco a un modelo más realista y complejo.

Otras líneas de investigación intentan centrarse en monitorizar la contaminación existente en una zona determinada [14, 16, 19]; relacionar la polución del aire con enfermedades o empeoramiento de la salud de seres vivos [15]; analizar la contribución e influencia de varias fuentes de contaminantes en zonas afectadas [13]; simulación y caracterización de incidentes químico-biológicos [7], entre otras.

2. Modelo de dispersión de contaminantes

La dispersión de contaminantes a través del aire es un fenómeno difícil de representar o de estudiar sin el uso de una herramienta que lo simplifique. Esto se debe a su complejidad y a la inmensa casuística que existe en sus escenarios (la mayoría de los casos en zonas industriales, urbanas o al aire libre), donde es necesario utilizar modelos que faciliten la representación o estudio de la dispersión de un contaminante. En la referencia [6] se puede observar los modelos de propagación de contaminantes que la Unión Europea describe y a los que hace referencia.

Por definición, un modelo matemático es un sistema compuesto por una o más ecuaciones que relacionan unas variables de entrada con unas variables de salida, que a su vez permiten representar un comportamiento o fenómeno determinado. Éstos a veces vienen restringidos por una serie de condiciones iniciales o de entorno que son útiles para explicar el fenómeno con una mayor simplicidad, como ocurre en la propagación de contaminantes en el aire.

Existen tres tipos de modelos: deterministas, empíricos/físicos y mixtos. En un modelo determinista, unas mismas entradas producirán unas mismas salidas, no se contempla ninguna aleatoriedad en los resultados; el matemático es un modelo de investigación científica que se basa en la experimentación, observación de fenómenos, análisis estadístico y lógica empírica; en modelos mixtos se combinan ambas técnicas. A su vez, dentro del grupo de modelos empíricos/físicos se pueden diferenciar dos grupos, los modelos que están basados en ecuaciones matemáticas y los modelos empíricos que están basados en experimentos donde se realizan medidas y que, conjuntamente con un análisis estadístico, permiten su definición.

A menudo, cuando se necesita obtener un modelo que represente un comportamiento, primero se deducen ecuaciones sencillas abordando un problema más simplificado con respecto a la realidad, obteniéndose de manera progresiva modelos de mayor complejidad.

Por último lugar, mencionar el método de propagación de contaminantes Gaussian Plume, que se utilizará a lo largo del Trabajo de Fin de Grado y que se describirá a continuación.

2.1 Modelo Gaussian Plume

En nuestro caso, el modelo Gaussian Plume es un modelo puramente matemático basado en la ecuación de convección-difusión, que asume un modelo simplificado de la propagación de contaminantes a través del aire.

Las hipótesis en las que se basa el modelo de Gaussian Plume son las siguientes [20]:

- El desarrollo del modelo se lleva a cabo para un estado estacionario.
- La difusión de la masa es despreciable en el eje X.
- La velocidad del viento es constante, se considera que las variaciones del viento son despreciables.
- La fuente de contaminación posee una tasa de emisión continua y constante.
- Hipótesis de “Tierra plana”.

Su objetivo es poder representar en una sola ecuación compacta todos los fenómenos químicos-físicos que sean posibles, como la velocidad del viento, temperatura ambiental, presión o reacciones químicas de los contaminantes con el medio.

Este modelo es de uso muy común debido a su simplicidad de implementación a nivel de simulaciones en entornos de desarrollo matemático, y es denominado como “Gaussian Plume” porque la distribución de los contaminantes se produce con forma de pluma, mientras que la distribución de concentración del contaminante en el interior de esta forma son de tipo gaussianas [véase Ilustración 1].

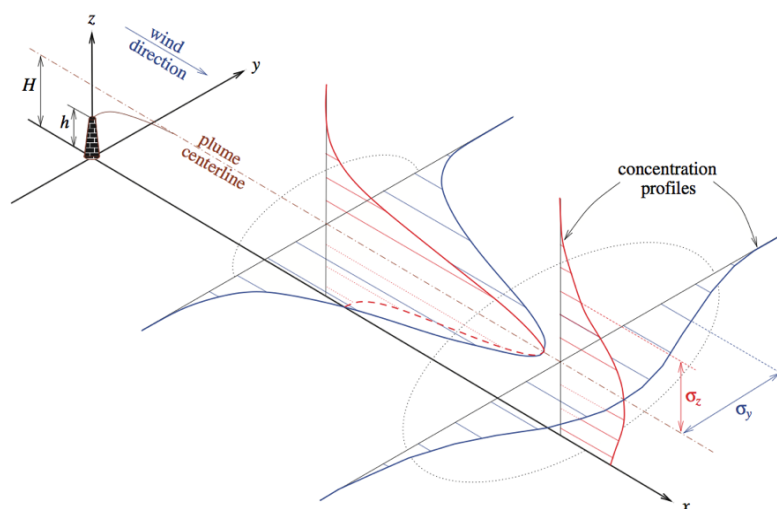


Ilustración 1: Modelo Gaussian Plume

Este modelo se describe matemáticamente por la siguiente ecuación, que es el resultado

de resolver la ecuación de convección-difusión mediante el método de elementos finitos:

$$C = \frac{Q}{2\pi U a_y a_z |x - x_s|^{b_y + b_z}} e^{\frac{-0.5(y - y_s)^2}{(a_y |x - x_s|^{b_y})^2}} \left(e^{\frac{-0.5(z - z_s)^2}{(a_z |x - x_s|^{b_z})^2}} + e^{\frac{-0.5(z + z_s)^2}{(a_z |x - x_s|^{b_z})^2}} \right)$$

Ecuación 1

El modelo se define para $x > x_s$, donde los parámetros son los siguientes:

C: Concentración en $\left[\frac{mg}{m^3}\right]$.

(x, y, z): Coordenadas en las que se quiere evaluar la función Gaussian Plume.

(x_s, y_s, z_s): Coordenadas en las que se localiza la fuente/s de contaminante/s.

Q: tasa de emisión de contaminantes [Kg/s].

U: velocidad del viento en la dirección \vec{x} en $\left[\frac{m}{s}\right]$.

a_y, b_y, a_z, b_z : parámetros que dependen de las condiciones atmosféricas, en función de la clase de estabilidad les corresponderá unos valores u otros [28].

Además, el modelo solo tiene sentido práctico cuando se evalúa matemáticamente en posiciones $x > x_s$. Esto es debido a la simplificación del modelo de propagación, en la que se asume que el contaminante solo se propaga en posiciones con coordenadas mayores a la posición de la fuente en el eje x debido al efecto del viento [véase la Ilustración 1].

Ventajas y desventajas del modelo

La principal desventaja de utilizar el modelo Gaussian Plume es la necesidad de hipótesis previas, ya que en ciertas situaciones resulta que éstas se convierten en limitaciones para el estudio de la propagación de contaminantes (viento constante y unidireccional, tasa de emisión de contaminantes constante); sin embargo, resultan necesarias para poder simplificar el modelo.

Por ejemplo, en un escenario real, es prácticamente imposible encontrar una velocidad y dirección de viento constante para un intervalo de tiempo prolongado; al igual que es difícil encontrar una fuente que emita contaminantes con una tasa constante.

Aun así, el modelo Gaussian Plume es muy útil porque, a pesar de ser un modelo simplificado de la realidad, permite representar de forma aceptable el comportamiento de

la difusión de contaminantes en el aire y aporta flexibilidad (debido a sus numerosos parámetros) necesaria para realizar simulaciones.

Usos

Actualmente el uso de simulaciones está cobrando real importancia, porque tiene un bajo coste adquirir las herramientas necesarias para llevarlas a cabo (ordenador, entorno de desarrollo, etcétera).

El modelo Gaussian Plume puede tener diferentes utilidades: monitorizar la emisión de contaminantes una vez conocida la tasa de emisión de la fuente, la velocidad y dirección del viento; realizar una estimación de contaminación en diversos puntos de una zona geográfica en la que no se poseen sensores (en situaciones en las que se conoce la tasa de emisión de la fuente, dirección y velocidad del viento); localizar fuentes de contaminantes en situaciones en las que se desconoce la posición del origen de la contaminación, etcétera.

3. Sensores y modelo de medidas

A continuación se explicará el modelo de medidas utilizado y también el tipo de sensores que se podrían usar para la localización de fuentes de contaminantes.

En este Trabajo de Fin de Grado se ha considerado una fuente de contaminantes de tipo industrial, suponiendo como contaminantes micropartículas de Zinc, debido al peligro que supone para la salud de los seres vivos.

El zinc se utiliza en para las siguientes actividades [26]:

- Galvanizado del acero.
- Aleaciones de metales.
- Recubrimiento de metales.
- Protector de metales.
- Galvanizar metales.
- Conductor de calor y electricidad.

El procesado del Zinc implica el uso de grandes fundiciones utilizando éstas grandes hornos. Éstos son fuentes de contaminación por emitir varios tipos de gases, entre ellos micropartículas de zinc, causando efectos nocivos en la salud humana y en otras especies [27]. Sin embargo se debe puntualizar que a pesar de haber sido elegido el zinc como contaminante, ésta no es la única opción, ya que este modelo matemático simulado permite considerar cualquier otro contaminante aéreo. Solo es fundamental conocer el contaminante en caso de implementar una red de sensores real, en cuyo caso se deberían elegir los sensores adecuados para detectar el contaminante-objetivo.

En cuanto a medidas, la concentración de contaminantes se utilizará en unidades de $\frac{kg}{m^3}$, ya que es la medida que se utiliza en la simulación de Matlab para la concentración de contaminantes que perciben los sensores.

Los sensores propuestos para la localización de fuentes podrían ser administrados por el fabricante Dräger [5], que ofrece una gran variedad de sensores cuyos niveles de sensibilidad resultan muy adecuados (0 partes por millón y una resolución de 0.01 ppm en los mejores casos). Según el tipo de contaminante se puede realizar la conversión de ppm a Kg/m^3 , dependiendo de las propiedades físicas de éste.

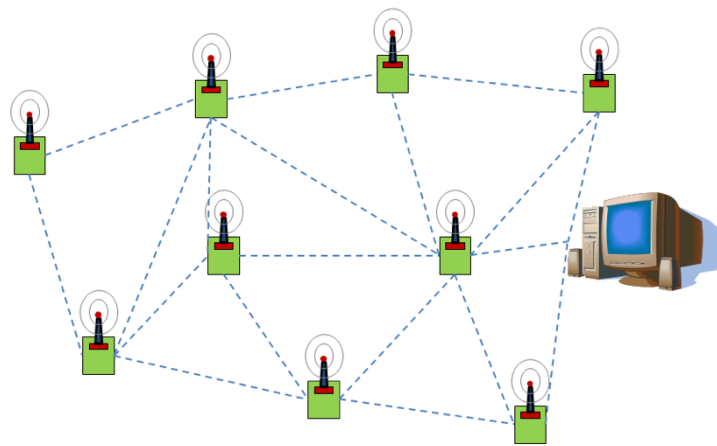


Ilustración 2: Ejemplo de una red de sensores

Como se verá posteriormente, el modelo de medidas utilizado en la segunda parte del estudio [Véase Página 38 - Localización de fuentes de contaminantes independientemente del modelo de propagación] será distinto, ya que en este caso se consideran las medidas de concentración de contaminantes en escala logarítmica. Esto es muy útil debido a que se ha considerado que los contaminantes en el aire se dispersan según el modelo Gaussian Plume, el cual depende en todas sus componentes de funciones exponenciales. Por tanto, utilizando la escala logarítmica, se simplifica mucho el modelo de medidas porque se convierte en una composición de ecuaciones lineales más sencillas.

Al no disponer de sensores reales desplegados en un escenario real, se decidió simular un escenario completo (con sensores y fuente) y modelar la emisión de contaminantes utilizando el modelo matemático Gaussian Plume. Dicho modelo recibe como entrada las posiciones de los sensores y la fuente, la velocidad del viento en la dirección del eje X y la tasa de emisión para obtener las medidas generadas de concentración de contaminantes que deberían mostrar los sensores.

4. Soluciones propuestas

Las soluciones propuestas intentan resolver el problema de la localización de fuentes de contaminación mediante el uso de algoritmos de optimización y de estimación.

La primera solución se basa en la localización de contaminantes mediante el uso del modelo Gaussian Plume. Se utiliza para simular la fuente de contaminantes y también en la optimización, ya que una buena elección de sus parámetros permite localizar la fuente minimizando el error cuadrático medio entre las medidas de los sensores y las medidas estimadas con el modelo optimizado.

Dado que dicha función es derivable, se pueden aplicar algoritmos de optimización basados en el uso del Gradiente de la función y de su matriz Hessiana para localizar la fuente.

En la segunda parte de la solución, se plantea encontrar una solución genérica para el problema, que sirva para localizar una fuente pero sin tener en cuenta el modelo de propagación que la caracterice (evitando la dependencia de parámetros que ello implica). De esta forma, solo se debería conocer: posiciones de los sensores y sus medidas de contaminación.

4.1 Algoritmos de optimización

Por definición, un algoritmo es una serie o conjunto de operaciones que permiten hallar la solución de un problema.

Un algoritmo de optimización es un algoritmo que permite seleccionar valores inmejorables para un conjunto de valores posibles de una función.

En este tipo de algoritmos se define una función matemática, por ejemplo f , que suele depender de una o varias variables, $f(x_1, x_2, x_3, \dots)$. Las variables (x_1, x_2, x_3, \dots) son las que habitualmente se desea optimizar.

Actualmente se conocen una gran cantidad de algoritmos de optimización, pero en el desarrollo de este trabajo de fin de grado se han utilizado los siguientes tres: método descendente del Gradiente Descendente o también denominado “Steepest descent” [10], método del gradiente adaptativo [4] y el método de Newton [17].

4.1.1 Método del Gradiente Descendente

El método del Gradiente descendiente es un algoritmo iterativo, cuya utilidad reside en localizar el mínimo de una función aproximándose a él iterativamente.

Este tipo de modelo se basa en el uso del gradiente de la función a minimizar y como tal, ésta tiene que ser derivable en los puntos del cálculo.

Antes de aplicar cualquier algoritmo de optimización sobre una función, ésta debe ser elegida correctamente para resolver el problema que se ha planteado.

Por ejemplo, se podría aplicar este método a una función lineal que represente el rendimiento de una máquina y que dependa de n parámetros, de tal forma, este método intentaría encontrar los parámetros adecuados para que el resultado de esa función evaluada en dichos valores sea mínimo.

Para que se comprenda mejor cómo actúan en general los algoritmos optimización, a continuación se explicará un breve ejemplo con una función cuadrática muy sencilla generada a través de un gráfico dinámico [29] a la que se le aplicará el algoritmo del Gradiente Descendente:

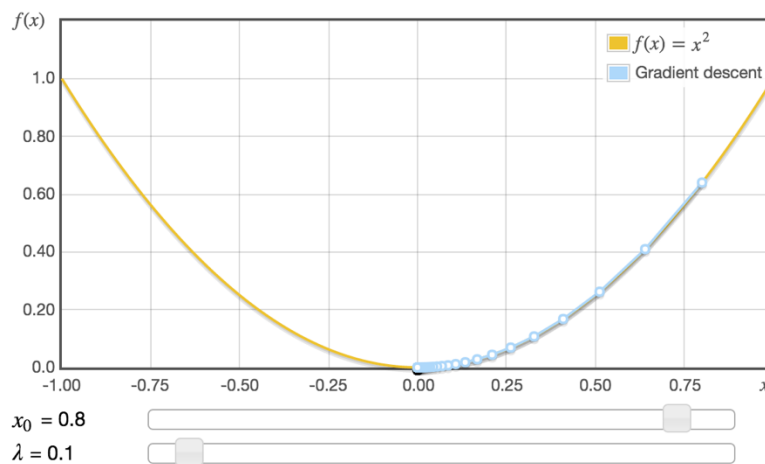


Ilustración 3: Ejemplo de método del Gradiente descendente sobre la función $f(x) = x^2$; $\lambda = 0.1$

En la función representada en la Ilustración 3 se quiere conocer cuál es el mínimo y además conocer para qué parámetros de los que depende la función (en este caso x) se obtiene el mínimo de dicha función.

La siguiente sentencia iterativa es la que define el algoritmo y se repetirá hasta llegar a la solución del problema:

$$x_{k+1} = x_k - \lambda \nabla f(x_k)$$

Ecuación 2

Donde:

x_k : son los valores numéricos que toman los parámetros de los que depende la función f (en la iteración k -ésima). x_k podría ser un vector de valores dependiendo de los parámetros de los que dependa la función f . Para la primera iteración, en el que x_k es x_0 , se tiene que definir x_0 con un valor inicial determinado a partir del cual iniciará el algoritmo.

x_{k+1} : son los valores numéricos que toman los parámetros de los que depende la función f en la iteración $k+1$ -ésima. Como se puede observar en el algoritmo, x_{k+1} se calcula iterativamente a partir de los valores que va tomando x_k en cada iteración.

λ : se denomina constante de aprendizaje, es un valor positivo, cuya función es que en cada iteración, la contribución del término $\lambda \nabla f(x_k)$ sea adecuada en relación al valor que tiene x_k (para que cada la variación entre los valores que va tomando x_k sea correcto).

$\nabla f(x_k)$: es el gradiente de la función f evaluada con los valores de x_k . La utilidad que tiene el uso del gradiente o derivada, es que proporciona en cada iteración la pendiente de la función, evaluada en los valores de x_k . De esta forma, el algoritmo cobra sentido y proporciona valores de la función f que para cada iteración debe cumplir $f(x_{k+1}) < f(x_k)$ para garantizar la estabilidad del algoritmo y que cada iteración garantice una aproximación hacia el valor mínimo de la función f , si se ha efectuado una correcta elección previa del parámetro λ . Posteriormente se verán distintos tipos de experimentos variando el valor que toma λ .

En la Ilustración 3, x toma el valor inicial 1 ($x_0 = 1$) y según aumentan el número de iteraciones la variación entre x_{k-1} y x_k es menor. Esto se debe a la contribución del gradiente (que refleja la intensidad de la pendiente de la función) que en la sentencia $x_{k+1} = x_k - \lambda \nabla f(x_k)$ es menor al aproximarse al mínimo.

A continuación se presenta otro ejemplo del uso del algoritmo de optimización del Gradiente descendente, pero con distinta variación en el parámetro λ .

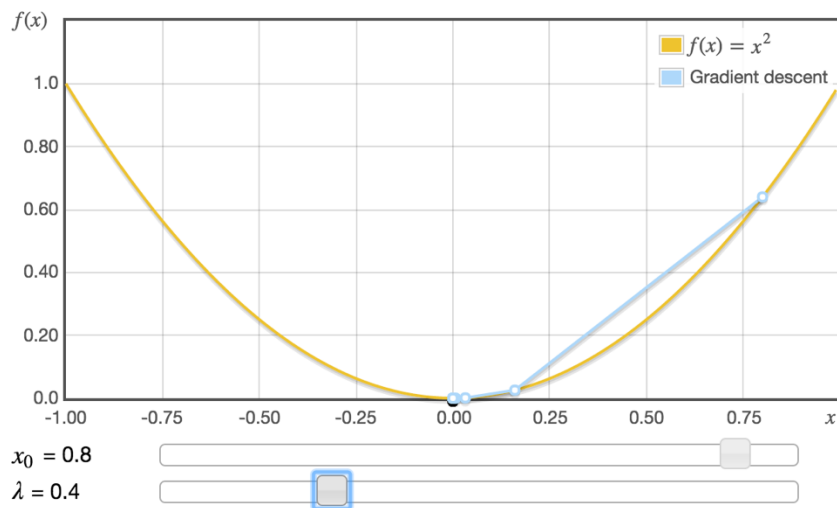


Ilustración 4: Ejemplo de método del Gradiente descendente sobre la función $f(x)=x^2$; $\lambda=0.4$

En la Ilustración 4 se puede observar el mismo experimento que en la Ilustración 3, pero con otro valor del parámetro λ . Lo primero a destacar es que el algoritmo consigue encontrar el mínimo de la función f en menos iteraciones. Esto es debido a que λ a tomado un valor mayor, que implica directamente una mayor contribución de $\lambda \nabla f(x_k)$, lo que provoca a su vez un decremento mayor en el cálculo de x_{k+1} para cada iteración.

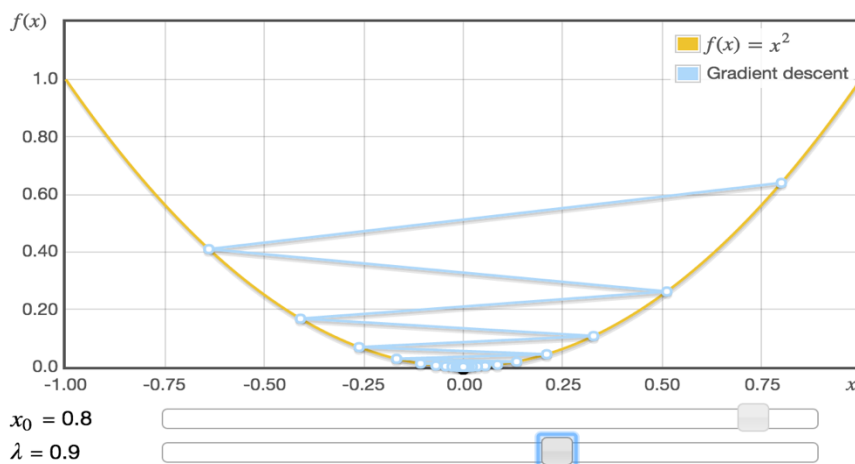


Ilustración 5: Ejemplo de método del Gradiente descendente sobre la función $f(x)=x^2$; $\lambda=0.9$

Sin embargo, en la Ilustración 5 el valor de λ es tan elevado que en las primeras iteraciones hace que el valor de x_{k+1} vaya oscilando alrededor del mínimo hasta llegar a la solución.

Por último, en la Ilustración 6, se puede observar que el valor de λ no es el adecuado, ya que no garantiza la estabilidad del algoritmo. Se puede apreciar como en cada iteración se va alejando de la solución correcta (encontrar el mínimo).

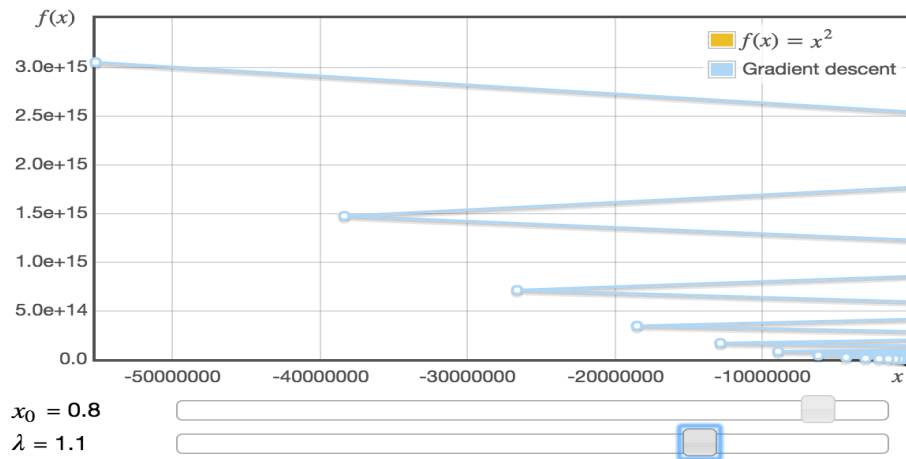


Ilustración 6: Ejemplo de método del Gradiente descendente sobre la función $f(x)=x^2$; $\lambda=1.1$

La conclusión después de los anteriores ejemplo es: el parámetro λ tiene que elegirse correctamente, para que el algoritmo alcance su objetivo. De ser éste más pequeño de lo necesario, el algoritmo tardaría más iteraciones en alcanzar la solución y de ser muy grande podría no alcanzarla nunca.

Los anteriores ejemplos son genéricos para una función cuadrática sencilla, pero en el apartado [Aplicación del método del Gradiente descendente para la localización de fuentes de contaminantes – Pág. 25] se mostrará cómo se utiliza el algoritmo del Gradiente Descendente para resolver el problema de la localización de fuentes de contaminantes.

Se ha descrito en profundidad este algoritmo debido a que los siguientes métodos utilizan también la información de la pendiente de la función a optimizar para llegar a la solución.

4.1.2 Método del Gradiente Adaptativo para la localización de fuentes de contaminantes

En la primera parte del estudio en este proyecto de fin de grado se comentó que se iban a utilizar 3 métodos de optimización para localizar la posición de una fuente.

El segundo algoritmo elegido es el Gradiente Adaptativo, porque a pesar de que aumenta su complejidad con respecto al Gradiente Descendente mejora sus prestaciones. Lleva a cabo una adaptación de la tasa de aprendizaje en cada iteración, lo que permite una mejora en la estimación de la posición de la fuente y una convergencia más rápida.

Para ello, la constante de aprendizaje se afecta por un término adicional (dependiente del gradiente de la función a minimizar, que continua siendo la misma). Así, el algoritmo del Gradiente Adaptativo, se describe matemáticamente como:

$$x_{k+1} = x_k - \frac{\lambda}{\sqrt{G_{kk} + \epsilon}} g_k$$

Ecuación 3

Donde

$$G_{kk} = g_k g_k^T$$

Ecuación 4

Donde

$x_k, x_{k+1}, \lambda, g_k$: parámetro con la misma definición que en el Gradiente Descendente [véase Ecuación 2].

G_{kk} : matriz diagonal, cuyos elementos son las componentes del gradiente de la función a minimizar (evaluado en la posición x_k) y elevadas al cuadrado.

ϵ : término que permite evitar indeterminaciones y divisiones entre cero a lo largo de la ejecución (normalmente del orden 10^{-8}).

En este tipo de algoritmos el valor de λ no cobra demasiada importancia, debido a que G_{kk} afecta en mayor medida al resultado final y se adapta en cada iteración.

4.1.3 Método de Newton para la localización de fuentes de contaminantes

Como los dos métodos anteriores, el método de Newton tiene el objetivo de optimizar una función dada.

El método de Newton básicamente utiliza la información que proporciona el gradiente y la matriz Hessiana de la función a minimizar, para así alcanzar la solución al problema de forma más óptima que en los algoritmos que se han visto hasta este punto.

La matriz Hessiana proporciona información sobre la curvatura de una función y ésta se aprovecha para mejorar la optimización.

Por definición, la matriz Hessiana de una función (que debe ser derivable en los puntos en los que se desea evaluar) es una matriz formada por las derivadas parciales de dicha función, y viene definida matemáticamente como:

$$\mathbf{F}(\mathbf{x}) = \begin{bmatrix} \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_2} & \frac{\partial^2 f(\mathbf{x})}{\partial x_2^2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_n} & \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_n} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_n^2} \end{bmatrix}$$

Donde

\mathbf{F} : Matriz Hessiana de la función $f(X)$

\mathbf{X} : vector de n componentes de las que depende la función $f(X)$

El método de Newton se define como:

$$x_{k+1} = x_k - \lambda F(x_k)^{-1} \nabla f(x_k)$$

Ecuación 5

Donde

$x_k, x_{k+1}, \lambda, f(x_k)$: [véase Ecuación 2].

$F(x_k)$: Matriz Hessiana de la función a minimizar (evaluada en x_k).

4.2 Escenarios de estudio

Los escenarios utilizados en este proyecto vienen definidos por:

- Área del escenario (definida por dos parámetros: ancho y largo [metros]).
- Número de sensores distribuidos aleatoriamente en el área (a una altura de 3 metros)
- Fuente de contaminación (encargada de liberar contaminación en el aire, con tasa de emisión constante y desde una altura de 10 metros).
- Velocidad y dirección del viento (velocidad constante de 10 m/s y dirección \vec{x})

En la Ilustración 7 se puede observar un ejemplo. Los sensores vienen definidos en color azul y rojo, y la fuente en color turquesa.

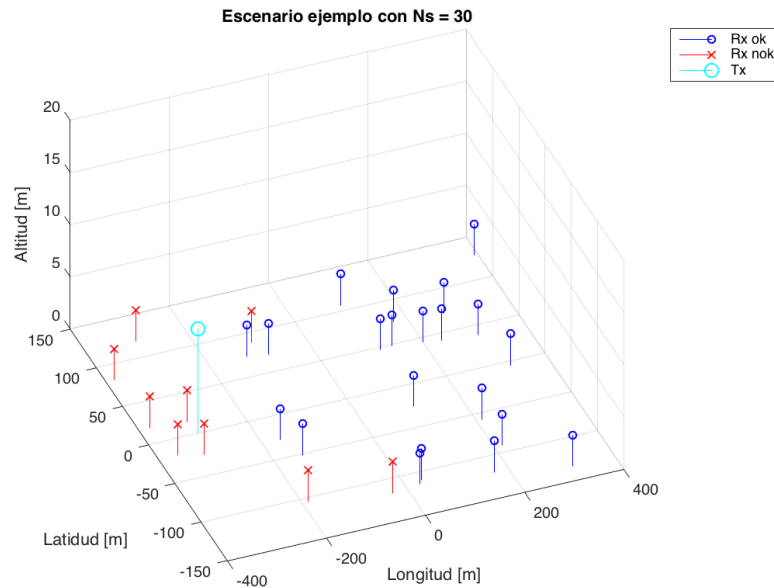


Ilustración 7: Ejemplo de escenario con 30 sensores

4.3 Localización de fuentes de contaminante basada en el modelo de propagación

A continuación se aplicarán los métodos de optimización descritos anteriormente para resolver el problema de localización de fuentes, además se comentarán los resultados obtenidos y se obtendrán algunas conclusiones.

Para la aplicación correcta de cualquiera de los métodos descritos, se deben considerar conocidos los siguientes parámetros (debido a que son los parámetros de los que depende el modelo Gaussian Plume):

- Posición exacta de los sensores.
- Posición de la fuente (estimada).
- Velocidad y dirección del viento.
- Tasa de emisión de contaminantes.

La posición de la fuente también se utiliza como parámetro del modelo Gaussian Plume, así que es necesario que se defina un valor inicial para éste al comienzo del algoritmo.

4.3.1 Aplicación del método del Gradiente descendente para la localización de fuentes de contaminantes

Para aplicar el método del Gradiente Descendente al problema de localización de fuentes, se necesita realizar un análisis previo en el que se defina la función matemática que se desea minimizar y si esta sería útil para resolver el problema planteado.

Si se consideran conocidos los parámetros anteriormente mencionados y se evalúan en el modelo Gaussian Plume, se puede obtener una función únicamente dependiente de la posición de la fuente (x_s, y_s, z_s) . De tal forma, cambiando la posición de la fuente se puede conseguir simular distintas distribuciones de contaminación en el aire.

Aplicando adecuadamente el método de optimización, se pueden obtener las posiciones aproximadas de la fuente de contaminantes. Para ello, en este Trabajo de Fin de Grado se ha intentado minimizar el Error Cuadrático Medio (MSE) entre las lecturas de contaminante en los sensores y las medidas de lectura que se obtendrían con una fuente localizada en las posiciones estimadas por el algoritmo (esta función a minimizar se denominará en adelante como “función de coste”).

La función de coste viene dada por la siguiente expresión:

$$f(x_k) = \sum_{i=1}^n (C(x_k) - \text{lecturas})^2$$

Ecuación 6

Donde

x_k : vector de coordenadas aproximadas de la fuente (x_s', y_s', z_s') en la iteración k-ésima del algoritmo.

$C(x_k)$: medidas que se obtienen a través del Modelo Gaussian Plume considerando la fuente de contaminantes en la posición (x_s', y_s', z_s') para la iteración k-ésima del algoritmo.

n : número de medidas o muestras proporcionadas por los sensores. En general una por cada sensor.

Lecturas: lecturas de concentración de contaminante reales proporcionadas por los n sensores de contaminación situados en el escenario de análisis.

El objetivo es situar la fuente ficticia en una posición en la que las medidas de contaminación que se obtengan a través del modelo Gaussian sean idénticas a las que nos proporcionan los sensores con las lecturas reales. Cuando esto se cumple, se supone que

la fuente ficticia se encuentra localizada justamente en la posición de la fuente real y el MSE es mínimo.

El problema de localización de fuentes al final se resume en encontrar unas coordenadas aproximadas (x_s', y_s', z_s') , tales que minimicen el error cuadrático entre las lecturas reales (sensores) y las lecturas generadas con el modelo Gaussian Plume considerando que la fuente se encuentra en las posiciones aproximadas (x_s', y_s', z_s') .

Por lo tanto, la Ecuación 2, en la que ya se encuentran todos los términos identificados se convierte en la siguiente:

$$x_{k+1} = x_k - \lambda \text{Grad} \left(\sum_{i=1}^n (C(x_k) - \text{lecturas})^2 \right)$$

Ecuación 7

A continuación es necesario calcular las derivadas parciales de la Ecuación 6 y evaluar dichas expresiones en la posición (x_s', y_s', z_s') , ya que se desea obtener las posiciones que minimizan el error cuadrático medio anteriormente mencionado.

Cálculo del Gradiente

Por definición, el gradiente de una función se calcula de la siguiente forma:

$$\nabla f(\mathbf{r}) = \left(\frac{\partial f(\mathbf{r})}{\partial x_1}, \dots, \frac{\partial f(\mathbf{r})}{\partial x_n} \right)$$

La función a derivar es $f(x_s, y_s, z_s) = \sum_{i=1}^n (C(x_s, y_s, z_s) - \text{lecturas})^2$ respecto a sus componentes x_s, y_s, z_s . La dimensión del vector del Gradiente será de 1x3, coincidiendo en este caso con el Jacobiano de la función $f(x_s, y_s, z_s)$.

A continuación se calcularán las derivadas parciales correspondientes a cada componente del Gradiente.

Aplicando la regla de la cadena sobre la Ecuación 6 para obtener sus derivadas parciales, se obtiene una expresión más compacta de la misma mediante cálculo matricial y aprovechando que Matlab es una herramienta muy eficiente para realizar cálculos sobre

matrices, se puede operar la ecuación de forma eficiente y evitar el cálculo del sumatorio, que por regla general utiliza muchos más recursos en nuestra herramienta de cálculo.

$$\text{Grad} \left(\sum_{i=1}^n (C(x_k) - \text{lecturas})^2 \right) = 2 \text{Grad}(f(x_s, y_s, z_s))(C(x_s, y_s, z_s) - \text{Lecturas})$$

Si se opera de forma matricialmente la expresión a la izquierda de la igualdad, se puede realizar el cálculo de una forma más simplificada, como se muestra a la derecha de la igualdad.

Hay que tener en cuenta que en la expresión anterior se realizan operaciones matriciales, en la que se tiene en cuenta la contribución de cada sensor y cada componente del gradiente para obtener como resultado final un vector de dimensiones 1x3 que será el encargado de rectificar en cada iteración el valor de x_k para que se aproxime en cada iteración al valor real de la fuente.

Como se puede observar, finalmente la expresión que se necesita derivar es $f(x_s, y_s, z_s)$ con respecto a las componentes x_s, y_s, z_s , de tal forma que se el gradiente de la función $f(x_s, y_s, z_s)$, cuya expresión completa puede visualizarse en el Anexo 1.

Una vez obtenida la expresión del gradiente de la función $f(x_s, y_s, z_s)$ se puede proceder a calcular iteración tras iteración el algoritmo visto en la Ecuación 7. Posteriormente se analizarán algunos resultados mostrando situaciones en las que el algoritmo alcanza la solución correcta y casos concretos en las que no la encuentra debido a algunas limitaciones.

Condiciones de parada del método

También se deben definir ciertas condiciones que permitan la detención del algoritmo, se denominan condiciones de parada. La primera está relacionada con el número máximo de iteraciones que se desea que se ejecute el algoritmo para intentar encontrar la solución, una segunda condición de parada se ha utilizado como entrenamiento del algoritmo, en el que se evalúa el error entre la posición real de la fuente y la posición aproximada que se obtiene en cada iteración del algoritmo; y una última condición que detiene el algoritmo cuando la diferencia entre las posiciones x_k y x_{k+1} es menor a 0.1 m, porque en este caso se concluye que el algoritmo ha alcanzado un mínimo (sea local o global, sea

la solución esperada o sea errónea). Entonces, cuando una de estas tres condiciones de parada se cumpla, el algoritmo se detendría.

Resultados

La Ilustración 8 es un ejemplo en el cual se aplica el algoritmo del Gradiente Descendente (fuente situada en la posición $[-300, 0, 10]$ metros) y veinte sensores repartidos en la superficie de estudio (el número de sensores a distribuir en el escenario se define al inicio del programa).

En la ilustración se pueden observar cuatro tipos de elementos en función del color en el que estén representados. La marca de color azul cian es la fuente de contaminantes, que en esta simulación se encuentra en la posición $[-300,0]$ metros; las marcas de color rojo azul corresponden con los sensores activos, se trata de sensores cuyas medidas de concentración de contaminación superan un umbral definido inicialmente; en color rojo se encuentran los sensores que han sido descartados por no obtener medidas superiores o iguales al umbral definido; en color verde se puede observar la posición aproximada en cada iteración del algoritmo, se puede comprobar como el resultado final es la posición de la fuente con una precisión de 20 metros aproximadamente (este valor de precisión se puede definir en el programa, actualmente es 20 metros).

En la Ilustración 9 se puede observar la misma información que en la Ilustración 8, pero además se puede observar la evolución de la componente correspondiente a la altura de la fuente en cada iteración que ejecuta el algoritmo.

Como resultado final, en la consola de Matlab se obtiene alguna información que se puede visualizar en la Ilustración 10. En esta información se pueden encontrar datos como la posición final aproximada de la fuente, el error entre las lecturas obtenidas con la posición aproximada de la fuente y las lecturas obtenidas con la posición real, el número de iteraciones en las que el algoritmo ha encontrado la solución minimizando el error que se describía en la Ecuación 6.

Gracias al uso de modelos matemáticos, en el programa se pueden realizar infinidad de variaciones en los parámetros de la simulación, que permiten estudiar el comportamiento de la dispersión de contaminantes y también qué configuraciones son óptimas para el algoritmo y cuales no.

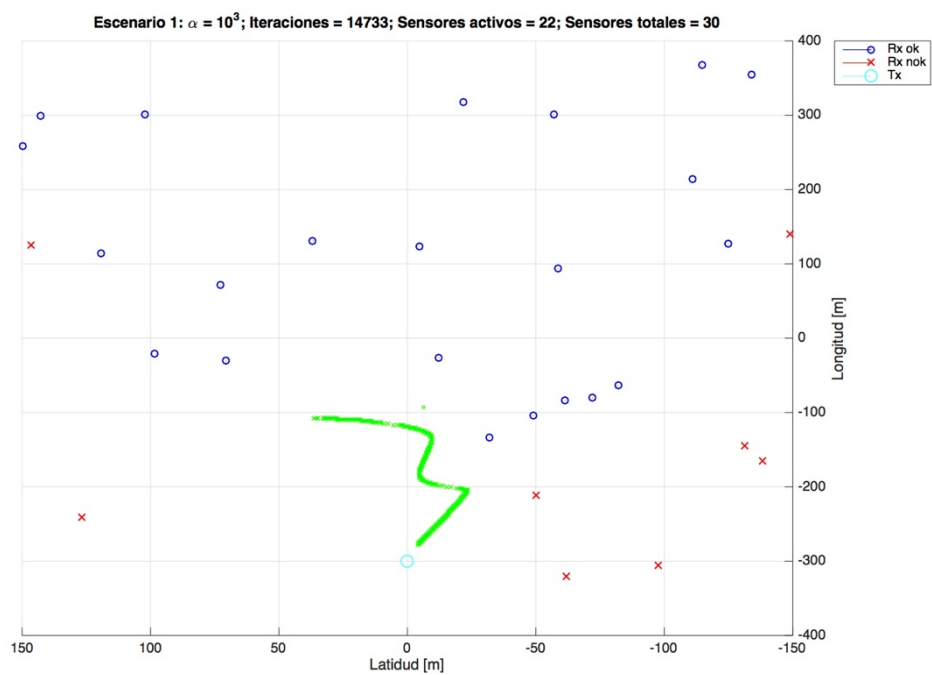


Ilustración 8: Escenario 1 - Aplicación del Gradiente Descendente (I)

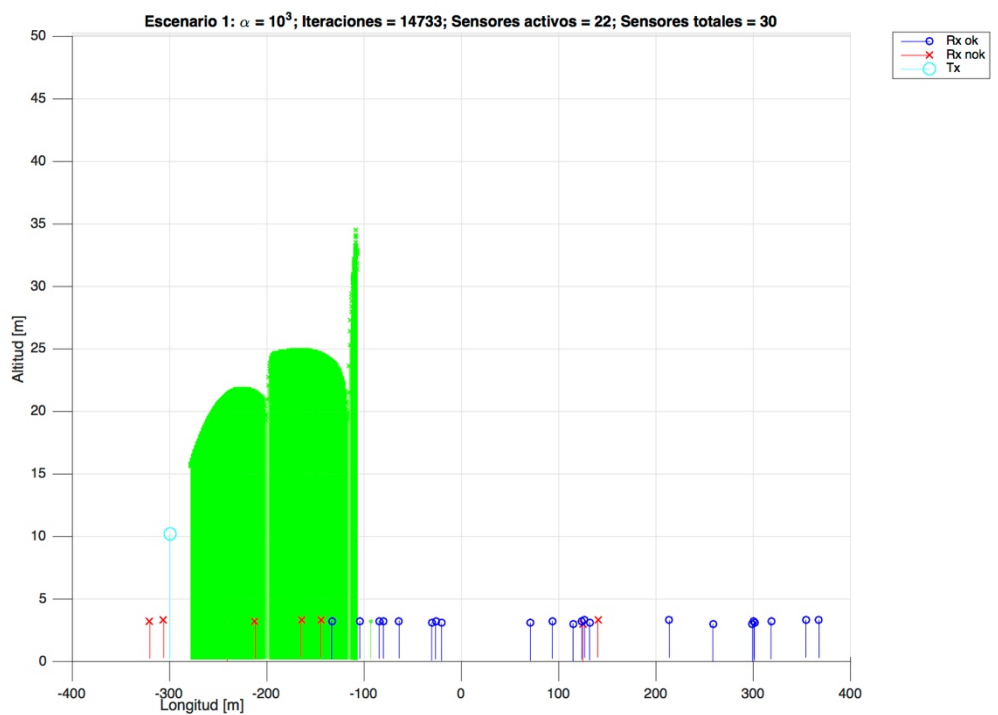


Ilustración 9: Escenario 1 - Aplicación del Gradiente Descendente (II)

```

pos_aprox =
    -280.4857    -1.8230     13.9747

error_lec =
    0.0072

Numero_sensores =
    10

|
iteraciones =
    1022

```

Ilustración 10: Resultado final de aplicación del Gradiente Descendente

En la Ilustración 11 se puede observar la evolución del algoritmo aplicado en el escenario presentado anteriormente como ejemplo, en ésta ilustración se compara el número de iteraciones que lleva a cabo el algoritmo frente al error entre la posición estimada de la fuente y la posición real de la misma.

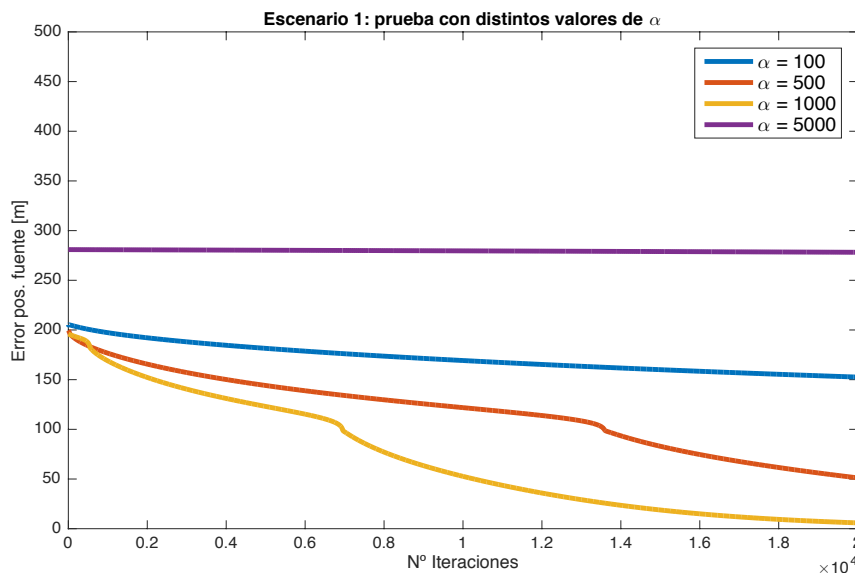


Ilustración 11: Resultados obtenidos para el escenario 1 cuando se aplican distintos valores de constante de aprendizaje

Se puede observar, que dependiendo del valor que tome la variable λ , o también denominada tasa de aprendizaje, se obtiene unos resultados u otro. Concretamente, en la se puede comprobar que a medida que λ incrementa su valor, el algoritmo por norma general converge más rápido a la solución, sin embargo, también se puede observar que si el valor de λ es demasiado grande (como es el caso de $\lambda = 5000$, el algoritmo se puede descontrolar no hallando la solución. Este fenómeno se explicó con ilustraciones más simples en la Ilustración 4, Ilustración 5 e Ilustración 6.

A continuación en la Ilustración 12 se muestra otro ejemplo en el que el algoritmo no alcanza la solución correcta debido a que cuando hay un número de sensores activos muy bajo, es más probable que el algoritmo tienda a encontrar mínimos locales de la función a minimizar. Esto se debe a que cuando hay pocas medidas que justificar mediante la posición estimada de la fuente, el número de soluciones posibles aumenta, ya que se tienen infinitas posiciones ficticias para situar la fuente de contaminantes. En este caso, cuando el algoritmo se detiene, muestra que se encuentra en un mínimo local debido a que se puede visualizar el error casi nulo entre las lecturas generadas por la fuente real y las lecturas generadas por la posición estimada por el algoritmo a pesar de que no coincide la posición estimada de la fuente con la posición real.

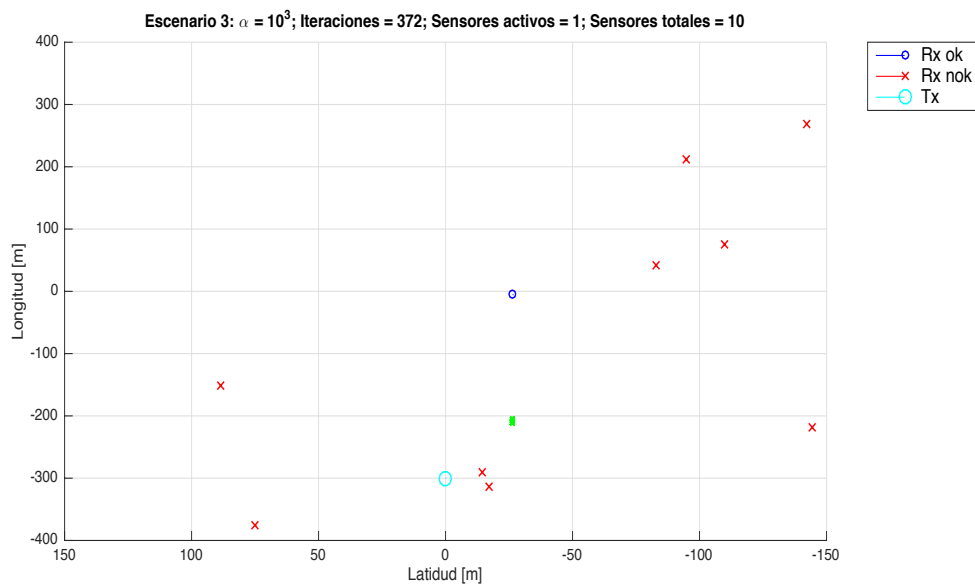


Ilustración 12: Escenario 2 - aplicación del método del Gradiente Descendente

Además, se pueden observar los resultados obtenidos a través de Matlab para dicho escenario, en el que el algoritmo ha encontrado una solución al problema de minimizar la función de coste pero no ha alcanzado la solución correcta porque la fuente real no está situada en la posición aproximada final que indica el algoritmo.

```
pos_aprox =
    -212.4586    -26.3776     29.5455

error_lec =
    1.8022e-05

Numero_sensores =
    1

iteraciones =
    372
```

También hay casos en los que el algoritmo no encuentre fácilmente la posición correcta de la fuente debido a unas lecturas muy bajas en los sensores, y dado este caso, las medidas que se obtienen mediante los sensores no son capaces de aportar la suficiente información al algoritmo.

De igual forma puede ocurrir que la definición del punto de partida del algoritmo, el valor que tomaría x_k en la primera iteración, no fuera el óptimo, por lo tanto el algoritmo podría detenerse al cumplir alguna de las condiciones de parada (por ejemplo, consumir el número de iteraciones máximo para intentar encontrar la solución). Este comportamiento se muestra en los resultados que se obtuvieron al ejecutar el algoritmo del Gradiente Descendente para el escenario que aparece en la Ilustración 13.

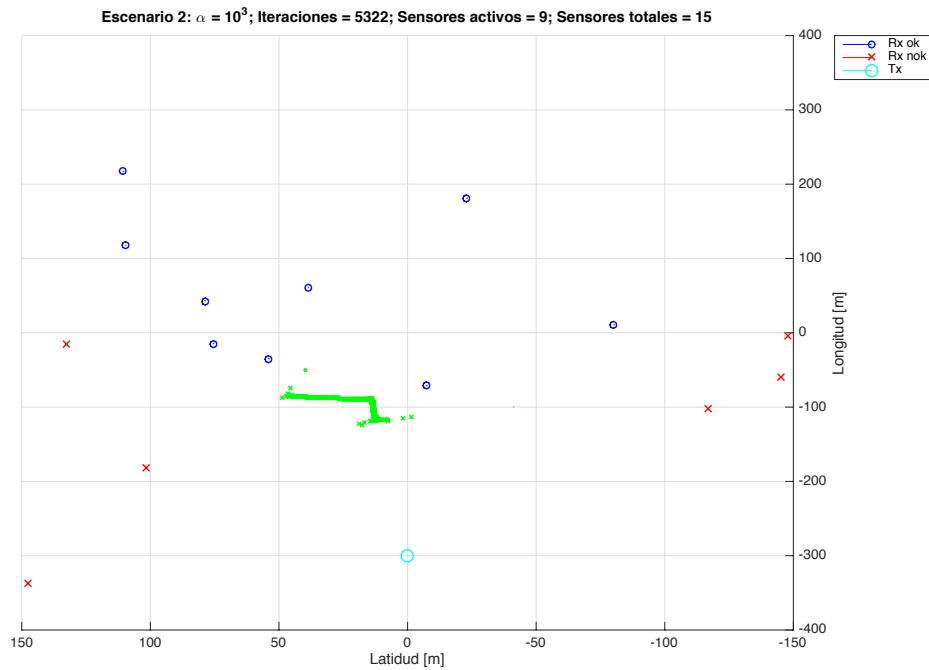


Ilustración 13: Escenario 3 - Aplicación del Gradiente Descendente

4.3.2 Aplicación del método del Gradiente Adaptativo para la localización de fuentes de contaminantes

Sustituyendo la Ecuación 2 en la Ecuación 3, se puede aplicar el método del Gradiente Adaptativo para la localización de fuentes, de nuevo minimizando el Error Cuadrático Medio entre las lecturas y las medidas aproximadas por el método.

Condiciones de parada del método

Las mismas descritas para el algoritmo Gradiente Descendente.

Resultados

A continuación se van a mostrar algunos resultados. En la Ilustración 15 se puede observar como se consigue localizar la fuente de contaminantes, mientras que en la Ilustración 13 se mostraba la aplicación del método del Gradiente Descendente sobre el mismo escenario y no conseguía encontrar la solución correcta. Este es un ejemplo en el que se pretende mostrar que el algoritmo del Gradiente Adaptativo es más fiable a la hora de localizar fuentes de contaminantes, ya que tiene una mejor respuesta frente a escenarios más complejos en comparación con el método del Gradiente Descendente. En este caso el aumento de la complejidad está justificado mediante la obtención de unos mejores resultados finales.

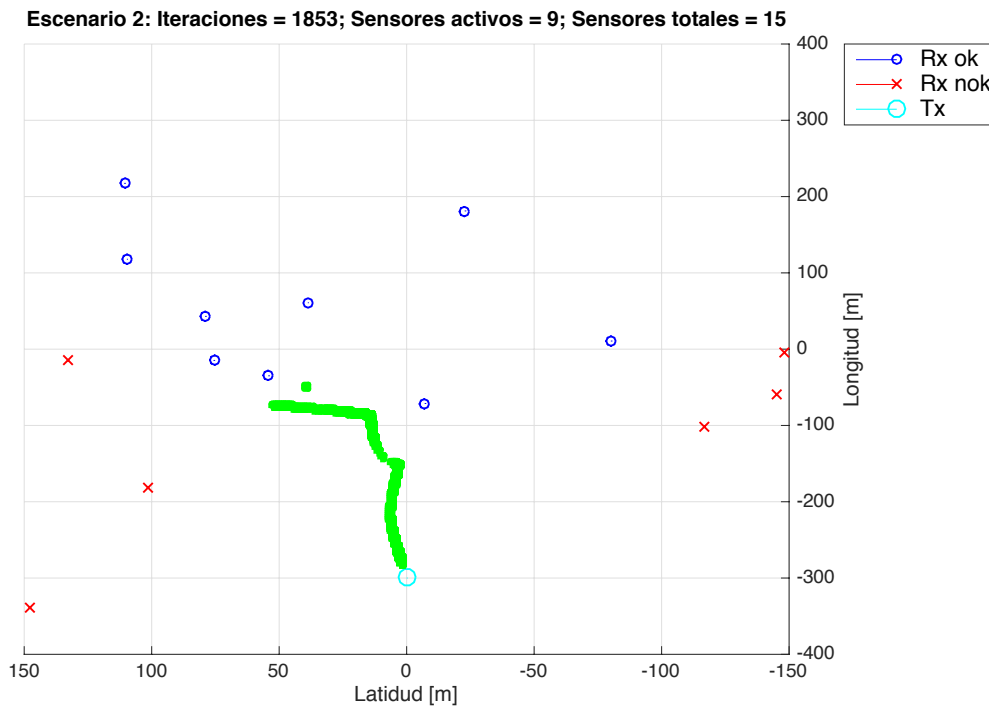


Ilustración 14: Escenario 3 - Aplicación del método del Gradiente Adaptativo

En los resultados mostrados se puede observar que se reduce en al menos un orden de magnitud el número de iteraciones necesaria para localizar la fuente (en comparación con el Gradiente Descendente).

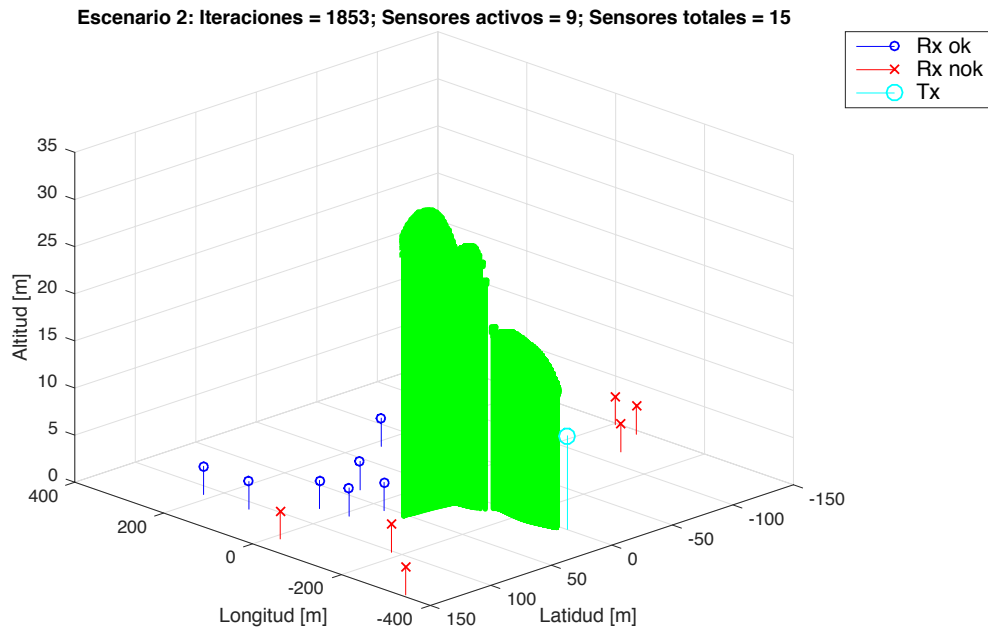


Ilustración 15: Escenario 2 - Aplicación del método del Gradiente Adaptativo

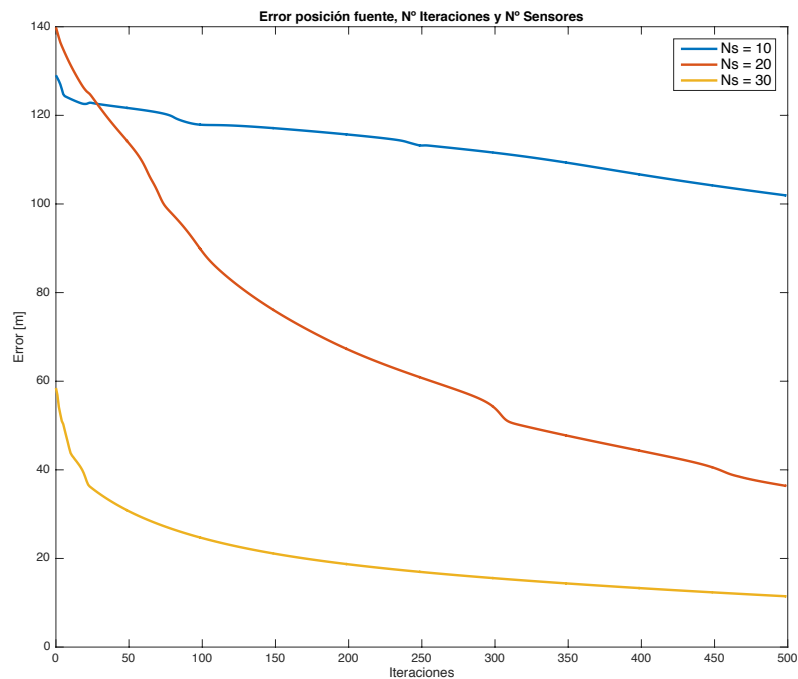


Ilustración 16: Evolución del error frente al número de sensores e iteraciones considerado

Casos en los que el algoritmo del Gradiente Adaptativo no suele encontrar la solución correcta:

- Cuando la posición inicial definida al inicio del algoritmo, x_0 , no es adecuada.

- Cuando el algoritmo encuentra algún mínimo local.
- Cuando las medidas que proporcionan los sensores son muy bajas o carecen de información útil para el algoritmo.

En la Ilustración 16 se muestra cómo varía el error en la aproximación de la fuente frente al número de iteraciones y número de sensores que se van a utilizar.

4.3.3 Aplicación del método de Newton para la localización de fuentes de contaminantes

Sustituyendo la Ecuación 2 en la Ecuación 5, se puede aplicar el método de Newton para la localización de fuentes de contaminantes.

Como se ha mostrado anteriormente en los métodos del Gradiente Descendente y el Gradiente Adaptativo, la función a minimizar será el error cuadrático medio (MSE) entre las lecturas de los sensores y las medidas de contaminación aproximadas mediante el método de optimización correspondiente. Hasta ahora, los métodos utilizaban como fundamento la información que proporciona la pendiente de la función a minimizar (evaluada en unos puntos concretos), de tal forma que permitía al algoritmo aproximarse al mínimo.

Condiciones de parada del método

Son las mismas condiciones de parada definidas para los anteriores algoritmos, aunque se añade una condición más (para garantizar la ejecución correcta del algoritmo): el valor de la inversa de la matriz Hessiana de la función de coste debe valer menos de $1e6$ ($F(x_k)^{-1} < 10^6$), ya que de ser mayor, el algoritmo se habrá descontrolado y dará resultados erróneos.

Resultados

A continuación se van a mostrar resultados de la ejecución del método de Newton como algoritmo de optimización para la localización de una fuente de contaminantes. Se podrán observar tipos de resultados como los vistos en los dos métodos de optimización anteriores.

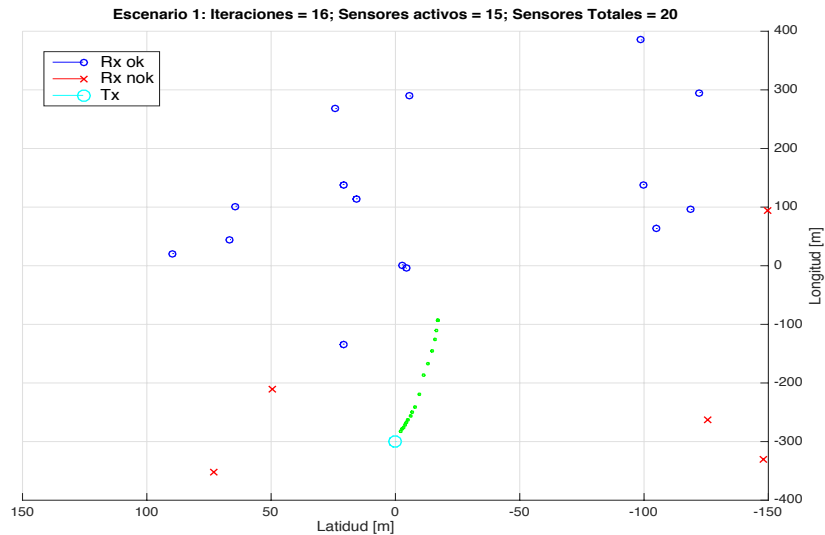


Ilustración 17: Escenario 1 - Aplicación del método de Newton

En la Ilustración 17 se muestran los resultados de aplicar el método de optimización de Newton a un escenario de una única fuente y una distribución de 20 sensores. Utilizando el método de Newton, bastan 16 iteraciones para localizar la fuente de contaminantes, resultado que sería imposible observar con los anteriores algoritmos, ya que necesitaban un mayor número de iteraciones (entre 2 y 3 ordenes de magnitud más) para localizar la fuente. Esto se debe a que el algoritmo de Newton permite a la constante de aprendizaje aprender de mejor forma y obtener mejores resultados. De nuevo el aumento de la complejidad del algoritmo se justifica con la obtención de unos mejores resultados.

En la Ilustración 18 se puede visualizar la evolución del error mientras se varía el número de sensores a lo largo de 30 escenarios distintos. Mientras mayor es el número de sensores, menos iteraciones son necesarias para localizar la fuente de contaminantes. Este comportamiento tiene sentido, ya que mientras más información se posea del comportamiento de la fuente (al tener un mayor número de sensores) más fácil será localizarla.

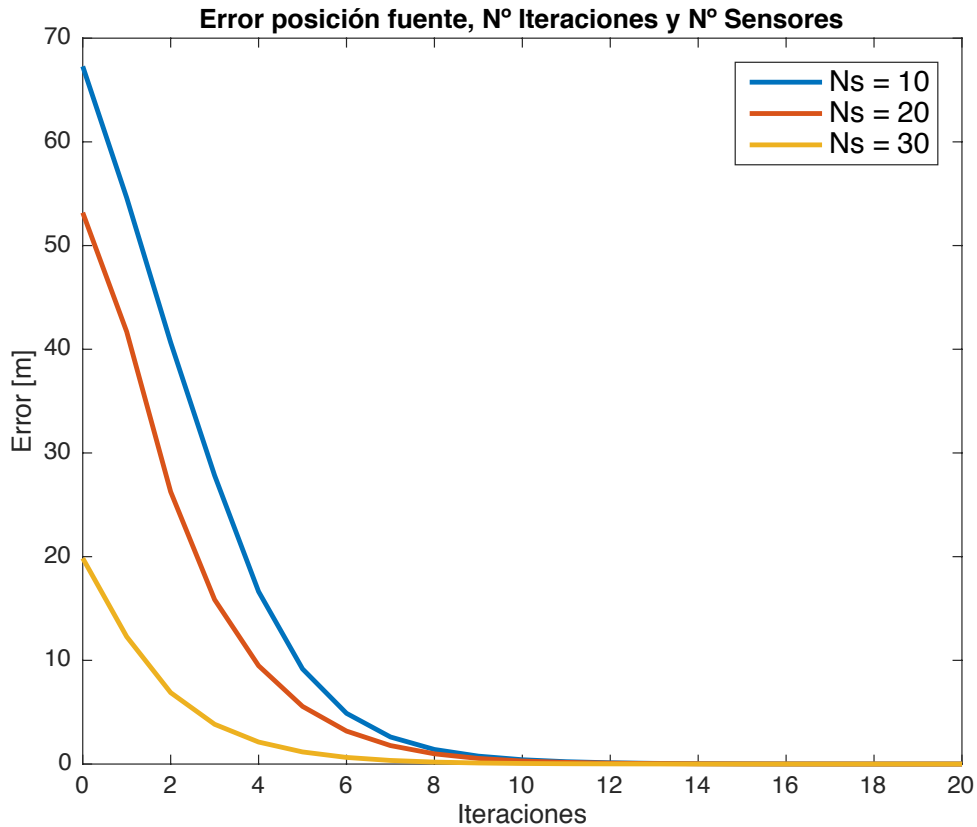


Ilustración 18: Error posición estimada de la fuente frente al número de sensores e iteraciones considerado

4.3.4 Conclusiones del uso de los métodos Gradiente Descendente, Gradiente Adaptativo y método de optimización de Newton

Hasta el momento se han utilizado tres algoritmos de optimización para intentar solventar el problema de la localización de fuentes de contaminantes en el aire. Los tres métodos de alguna manera estaban relacionados con el gradiente de la “función de coste”, aprovechan la información de la pendiente y curvatura (en el caso del método de Newton) de dicha función para alcanzar el objetivo de minimizar el MSE entre las lecturas y las medidas aproximadas.

Las condiciones de parada definidas para los algoritmos han funcionado correctamente, proporcionando resultados útiles de analizar y desechando los que no eran útiles.

A lo largo de las simulaciones se ha podido comprobar que a medida que el algoritmo de optimización se volvía más complejo, también mejoraban los resultados obtenidos (disminuyendo el número de iteraciones necesarias para alcanzar el objetivo y reduciendo el número de sensores necesarios para obtener un error menor).

También se ha podido observar que el número de sensores que se hallen desplegados sobre un escenario de estudio está ligado directamente con el número de iteraciones que necesita cada algoritmo para alcanzar la solución y eso está ligado directamente con el tiempo total en el que se alcanza la solución. Dicha relación ayudaría a elegir cuantos sensores necesito distribuir en una zona determinada, en función del número de iteraciones que quiero que mi algoritmo tarde en encontrar la solución, es decir, el tiempo en el que quiero que mi algoritmo localice a la fuente de contaminantes. Sin embargo, se debe de tener en cuenta que al aumentar el número de sensores también aumenta la complejidad de los cálculos que se realizan, por lo que se debería establecer un compromiso entre las dos variables (número de sensores y tiempo de ejecución) para que los resultados finales fuesen los esperados.

Gracias al anterior estudio, también se ha podido comprender que la posición inicial en la que empiezan los algoritmos de optimización es fundamental, ya que de no ser elegida correctamente, los algoritmos de optimización no logran localizar la fuente correctamente. Este tema es muy delicado cuando consideramos que los contaminantes se propagan según un modelo de Gaussian Plume, ya que la distribución a lo largo del eje y es mucho más estrecha que la distribución del contaminante en el eje x, y esto podría dar lugar a que el algoritmo empiece en una posición inicial en la que el no progrese como es debido.

Por último, la mayor limitación del anterior estudio se encuentra sin duda relacionada con las hipótesis previas necesarias al utilizar el método Gaussian Plume, como por ejemplo, conocer el modelo de propagación del contaminante y su tasa de emisión. Dicha hipótesis es muy limitante, porque en el mundo real sería muy complicado considerar un escenario en el que una fuente emite contaminantes y conocer cómo es su modelo de propagación.

4.4 Localización de fuentes de contaminantes independientemente del modelo de propagación

De la necesidad de independizar el estudio de un modelo de propagación concreto, surge la siguiente propuesta: encontrar un método de localización de fuentes que no dependa de un modelo de propagación concreto, éste debe ser lo más genérico que se pueda y proporcionar flexibilidad ante cualquier parámetro de los que dependa.

A continuación se va a desarrollar la segunda solución propuesta en este Trabajo de Fin de Grado, con el objetivo de resolver el problema de la localización de fuentes de contaminantes sin tener en cuenta el modelo de propagación de contaminantes.

La solución propuesta consta de un algoritmo de optimización mediante la técnica de mínimos cuadrados, y de nuevo se vuelve a repetir la estructura encontrada en los algoritmos Gradiente Descendente, Gradiente Adaptativo y método de Newton, pero esta vez contamos con un algoritmo que no necesita conocer el modelo de propagación del contaminante, ni la tasa de emisión de contaminantes de la fuente.

Para ello se utilizó la función “lsqcurvefit” implementada en Matlab, que se encuentra en la “Toolbox de Optimización”, recibe como parámetros: una función matemática definida por el usuario, una serie de variables de las cuales va a depender la función que se introduzca previamente, unos valores de entrada (las posiciones de los sensores) y unas salidas que se obtienen mediante observaciones (las lecturas de contaminación de los sensores).

Una vez definidos adecuadamente los parámetros de entrada de la función de optimización, el algoritmo intentará hallar los valores óptimos de los parámetros de los que depende la función de entrada para minimizar el error cuadrático medio (MSE) entre las observaciones introducidas (lecturas de contaminación en los sensores) y los valores que toma la función matemática (evaluada para los parámetros ajustados que proporciona el algoritmo como salida).

Es decir, el algoritmo va a ajustar la función que recibe de entrada para que justifique las medidas que le estamos proporcionando para unas observaciones dadas.

4.4.1 Optimización de la función FRG para localizar fuentes de contaminantes

Como se desea evitar una dependencia de un modelo de propagación concreto, se va a intentar definir una función matemática que sea útil para casos genéricos, que permita satisfacer las necesidades que surgen a la hora de intentar localizar fuentes de contaminación, pero sin conocer previamente el modelo de propagación de contaminantes.

Las hipótesis que se siguen tomando en cuenta con respecto a la primera parte del estudio son:

- Viento en una sola dirección y velocidad constante.
- Tasa de emisión de la fuente constante.
- Tierra plana.

Para intentar encontrar dicha función genérica que nos ayude a localizar fuentes de contaminantes, en los inicios del estudio se plantearon las siguientes preguntas:

- ¿Qué función se debe elegir para intentar localizar fuentes de contaminantes? Había que elegir una función que ajustada mediante sus parámetros permita interpolar el resto de la función gracias a los puntos sobre los que se tiene referencia (posiciones y medidas de los sensores).
- ¿Qué características debe tener?
- ¿Qué limitaciones tendría?

Gracias a la primera parte del estudio [véase apartado Localización de fuentes de contaminante basada en el modelo de propagación – Página 24], en el que se analizó el comportamiento de la distribución de contaminantes en el aire mediante el modelo Gaussian Plume, se pudieron identificar una serie de características que serían esenciales para la función a optimizar por el algoritmo:

- $x_c < x < x_{\max}$: la función debe estar limitada en el eje X. La función debe ser nula para todas las posiciones con componente X menores a la de la fuente (debido a los efectos del viento que seguimos considerando que sopla en una sola dirección y con velocidad constante) y tener un alcance máximo (debido a que la concentración del contaminante disminuye a medida que nos alejamos de la fuente y llegaría el momento en el que dicha concentración sea nula).
- $-y_{\max} < y < y_{\max}$: la propagación de contaminantes tiene simetría en el eje Y si el viento sopla en la dirección del eje X y con velocidad constante. Al igual que en el eje X, la función tiene que tener un alcance máximo que viene dado por la variable y_{\max} . Además, la pendiente en la componente Y debe de ser mayor que la pendiente en la componente X, debido a que el viento sopla en dirección X y consigue que los contaminantes se propaguen más lejos. Este último comportamiento se pudo observar mediante el análisis de la distribución de contaminantes a través del modelo Gaussian Plume en la primera parte del Trabajo de Fin de Grado.
- $0 < z < h + h_0$: dado que el objetivo es llevar a cabo una interpolación del modelo real de contaminantes para localizar la fuente, la altura de la función FRG

va a representar la concentración de contaminantes en unidades logarítmicas para una posición dada (x, y) del espacio.

- La función debe ser lineal, debido a que se va a trabajar con un modelo de medidas logarítmicas. Se define así porque se pudo observar en el modelo de Gaussian Plume que la pendiente de un modelo de propagación de contaminantes decae en función de expresiones exponenciales (tanto para el eje X como para el eje Y), de tal forma que si se realizan todos los cálculos con unidades logarítmicas, se podrá trabajar de una forma más sencilla, ya que se trabajaría con funciones lineales.

Construcción de la función FRG

Como resultado del anterior análisis de requisitos, se decidió construir una función para que pudiese satisfacer las necesidades tan específicas del problema a abordar. Se denominó “Función FRG” y su expresión matemática es:

$$FRG = \frac{h}{h + \epsilon} \left(\frac{x_c - x}{x_{max}} + 1 \right) \left(-\frac{|y - y_c|}{y_{max}} + 1 \right) + h_0$$

Ecuación 8: Función FRG

Donde:

h : altura relativa de la función.

h_0 : base de la función. A partir de esta altura se desarrolla la función, su función es hacer de offset.

x : coordenada X en la que se evalúa la función FRG.

x_c : coordenada X del centro de la función FRG.

x_{max} : alcance máximo de la función FRG en el eje X.

y : coordenada Y en la que se evalúa la función FRG.

y_c : coordenada Y del centro de la función FRG.

y_{max} : alcance máximo de la función FRG en el eje Y.

ϵ : constante que evita indeterminaciones. Definida en el orden de 1e-8.

En la Ilustración 19, Ilustración 20 e Ilustración 21 se puede observar un ejemplo de la función FRG para unos parámetros concretos. Se puede comprobar que cumple con los requisitos definidos previamente en el análisis.

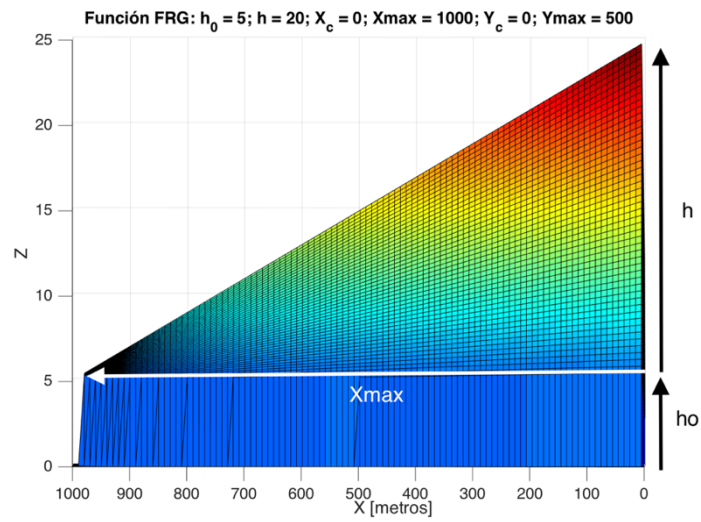


Ilustración 19: Función FRG (I)

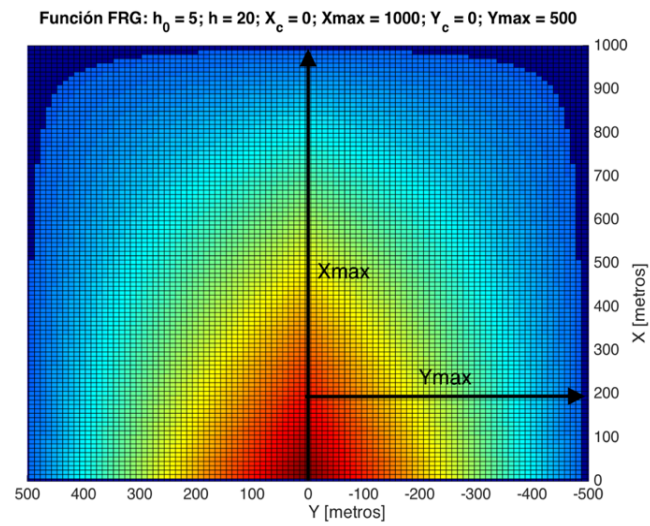


Ilustración 20: Función FRG (II)

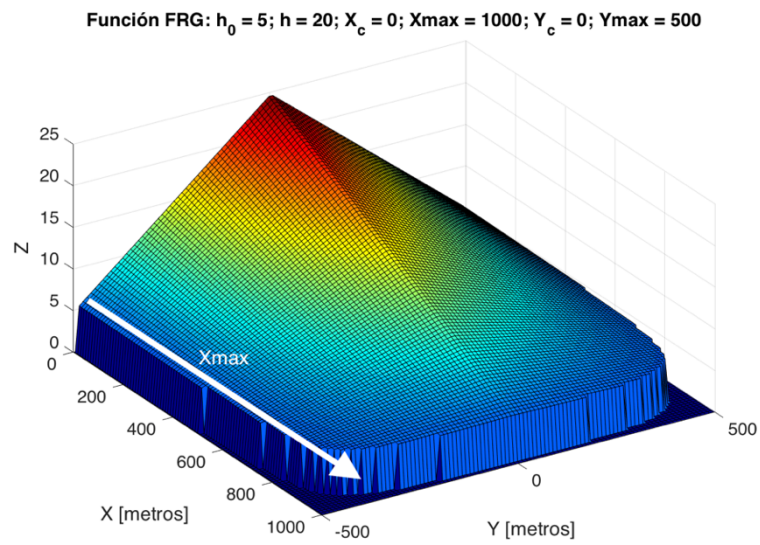


Ilustración 21: Función FRG (III)

Para llevar a cabo la optimización de la función FRG de forma correcta debe quedar claro cuáles son los pasos a seguir y el objetivo final con el que se realiza la optimización.

En primer lugar, los parámetros de la función FRG que se desean optimizar mediante el algoritmo son $x_c, x_{max}, y_c, y_{max}, h$ y h_0 . A su vez, el algoritmo tiene en cuenta las lecturas de los sensores y sus posiciones para realizar la optimización de los parámetros anteriormente mencionados. Se va a intentar minimizar el error cuadrático medio entre la salida generada con la función FRG y las lecturas que muestran los sensores, es decir, se va a construir una función FRG para que justifique las medidas proporcionadas por los sensores. modelo de propagación.

Resultados

A continuación se mostrarán los resultados de aplicar el algoritmo de optimización sobre la función FRG.

Al igual que en la primera parte del estudio, la simulación de medidas de concentración en los sensores se lleva a cabo utilizando la función Gaussian Plume, porque no se disponen de sensores reales que nos proporcionen medidas. El modelo Gaussian Plume no aporta ninguna información adicional a las medidas de concentración, simplemente sirve para simular una fuente de contaminantes.

En la Ilustración 23, Ilustración 22 e Ilustración 24 se puede observar la función FRG ajustada adecuadamente para intentar reconstruir el modelo de propagación del contaminante y localizar la fuente.

Como se puede observar se obtiene una muy buena aproximación para la mayoría de las medidas (aproximar las medidas con alta concentración es más prioritario para el método de estimación) y además localiza la fuente con gran precisión.

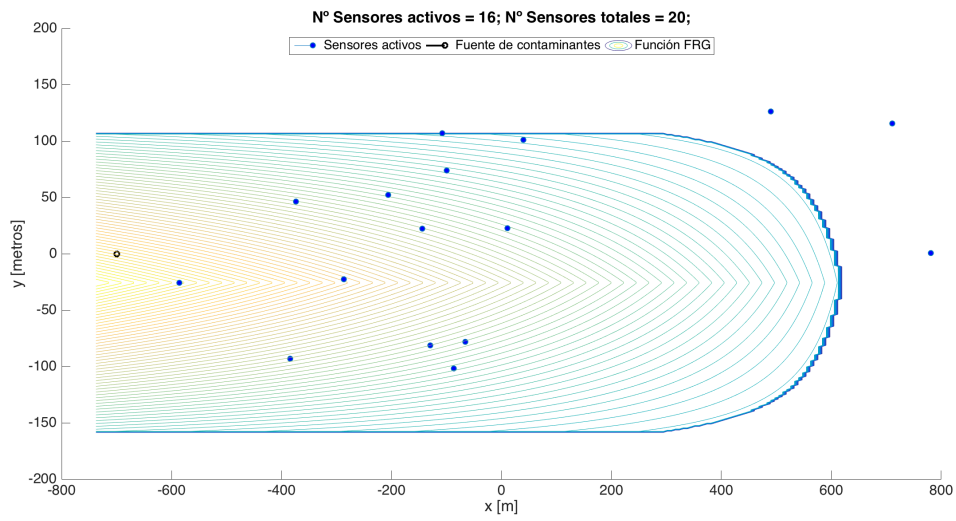


Ilustración 23: Optimización de función FRG para localización de fuentes (I)

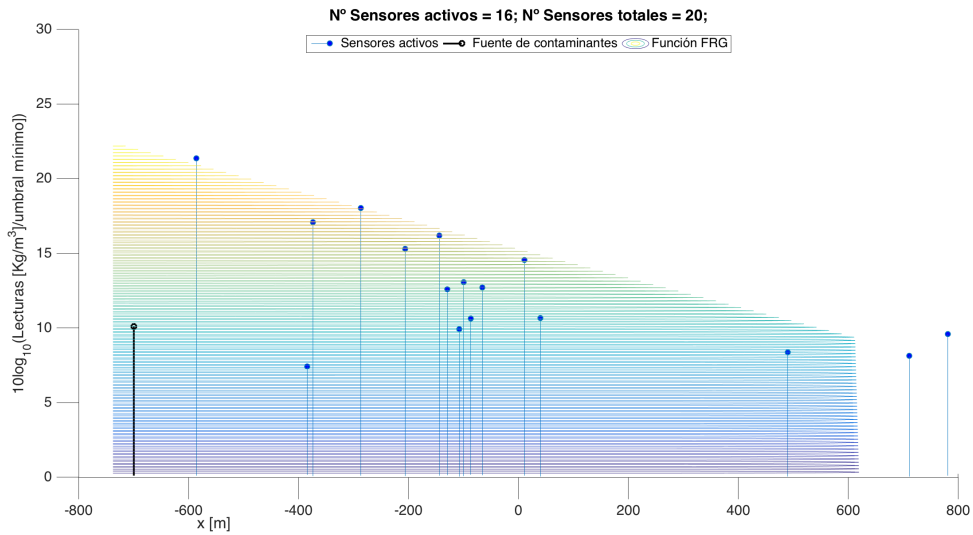


Ilustración 22: Optimización de función FRG para la localización de fuentes (II)

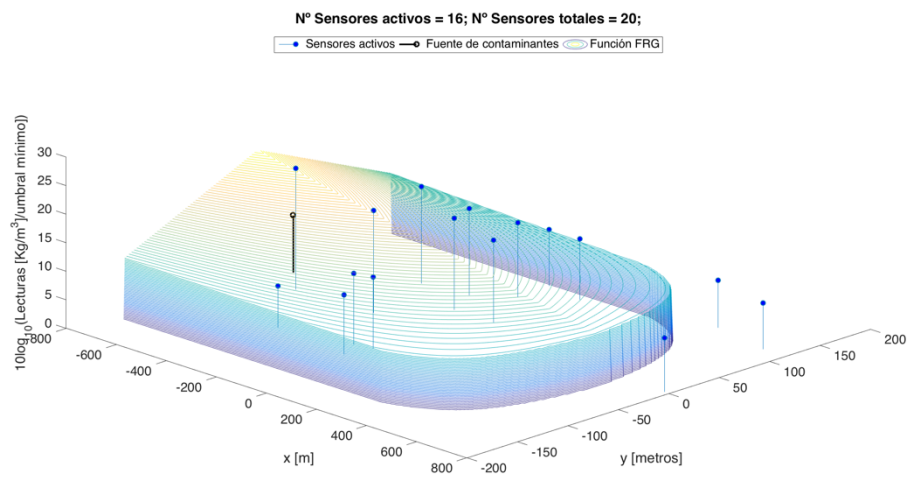


Ilustración 24: Optimización de función FRG para localización de fuentes (III)

Resultados considerando errores

Los errores de medida en los sensores que se consideraron en este proyecto fueron:

- Error de offset: error que afecta todos los sensores de la misma forma, sumando un valor constante a las medidas de concentración.
- Ruido blanco gaussiano: afecta aleatoriamente a las medidas que capturan los sensores (depende de parámetros como media y varianza).

A continuación se muestran los resultados de aplicar valores distintos de error de offset:

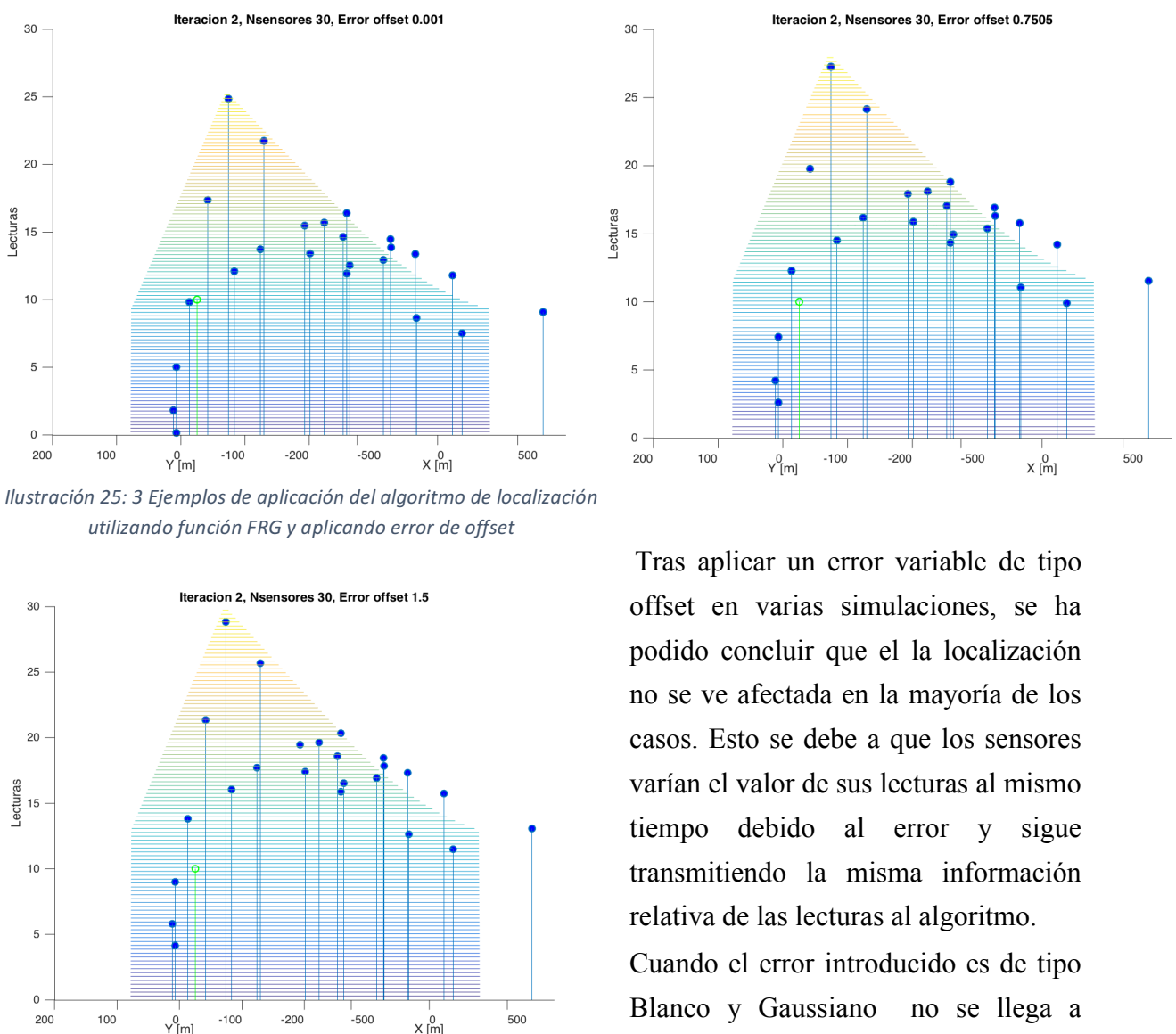


Ilustración 25: 3 Ejemplos de aplicación del algoritmo de localización utilizando función FRG y aplicando error de offset

Tras aplicar un error variable de tipo offset en varias simulaciones, se ha podido concluir que la localización no se ve afectada en la mayoría de los casos. Esto se debe a que los sensores varían el valor de sus lecturas al mismo tiempo debido al error y sigue transmitiendo la misma información relativa de las lecturas al algoritmo.

Cuando el error introducido es de tipo Blanco y Gaussiano no se llega a ninguna conclusión precisa, ya que se

aprecia demasiada diversidad de resultados. En algunas simulaciones el ruido favorece la localización de la fuente y en otras la empeora.

4.4.1.1 Método para localizar dos fuentes de contaminantes

Hasta ahora en este Trabajo se ha resuelto el problema de localizar una fuente de contaminantes, pero a continuación se va a intentar resolver el problema de localización de 2 fuentes de contaminantes.

En esencia, la base de esta solución es la misma que para la localización de una única fuente, con la diferencia de que ahora ejecutaremos el algoritmo de optimización dos veces. La primera ejecución permitirá localizar una primera fuente de contaminantes y con la segunda iteración se intentará localizar la segunda fuente.

En la primera ejecución se intenta minimizar el error cuadrático medio entre las lecturas de contaminación halladas en los sensores y los valores que devuelve de salida la función FRG una vez ha sido optimizada. De esta forma, si existiesen dos fuentes en el escenario de análisis, el algoritmo intentará minimizar el error cuadrático medio de todas las medidas que le estamos introduciendo de entrada. Como se está utilizando el ajuste de la función FRG para minimizar el MSE, resulta imposible éste minimice el error utilizando una sola función, debido a que si las fuentes de contaminación están lo suficientemente separadas, las medidas de concentración de contaminantes decaen en el espacio intermedio entre las fuentes y obviamente son máximas en las posiciones en las que se encuentran las fuentes. Por este motivo, lo que se realizó fue un primer análisis para localizar una sola fuente con la función FRG y un segundo análisis para minimizar el error residual con el resto de las medidas que no se lograron ajustar en la primera iteración.

En la Ilustración 26 e Ilustración 27 se puede observar el resultado final de aplicar el algoritmo de optimización junto con el uso de la función FRG. Se logra localizar las dos fuentes correctamente utilizando un método de optimización que permita ajustar una función genérica FRG para reconstruir en gran medida el modelo de dispersión del contaminante.

En la Ilustración 27 se puede observar desde otra perspectiva como las dos funciones FRG desplegadas sobre el escenario intentan ajustarse todo lo posible a las medidas de concentración de contaminante que los sensores proporcionan.

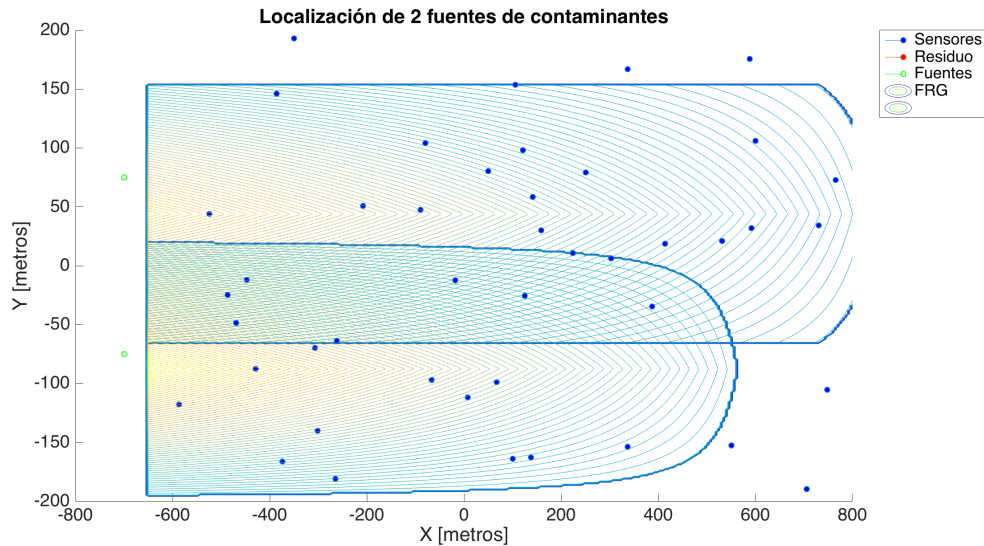


Ilustración 26: Localización de dos fuentes mediante ajuste de función FRG (I)

En la Ilustración 28 e Ilustración 29 se observa otro ejemplo de localización de fuentes de contaminación en caso de que las fuentes no estén alineadas pero sí suficientemente separadas para que el algoritmo separe las dos funciones FRG lo suficiente, y permita reconstruir los modelos de dispersión de contaminantes correctamente.

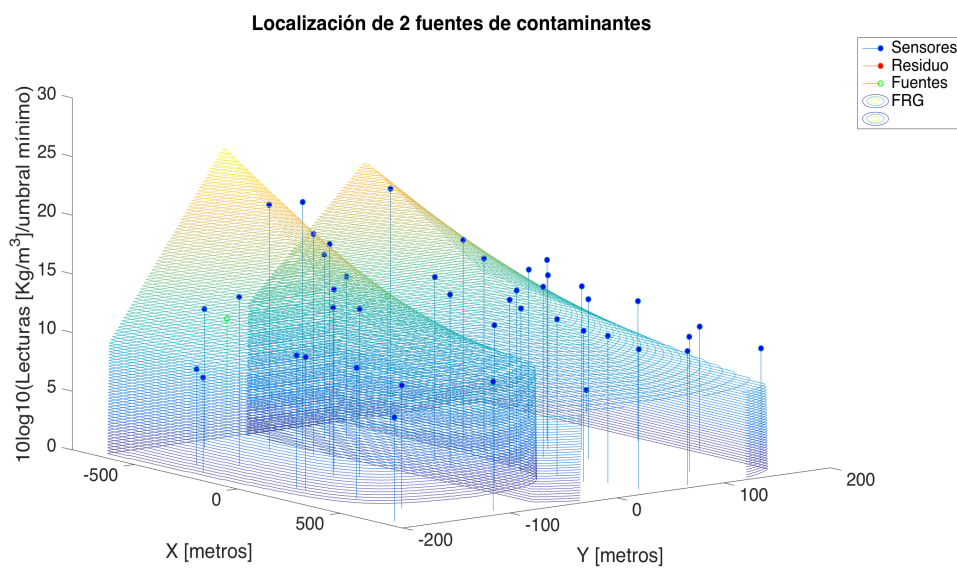


Ilustración 27: Localización de dos fuentes mediante ajuste de función FRG (II)

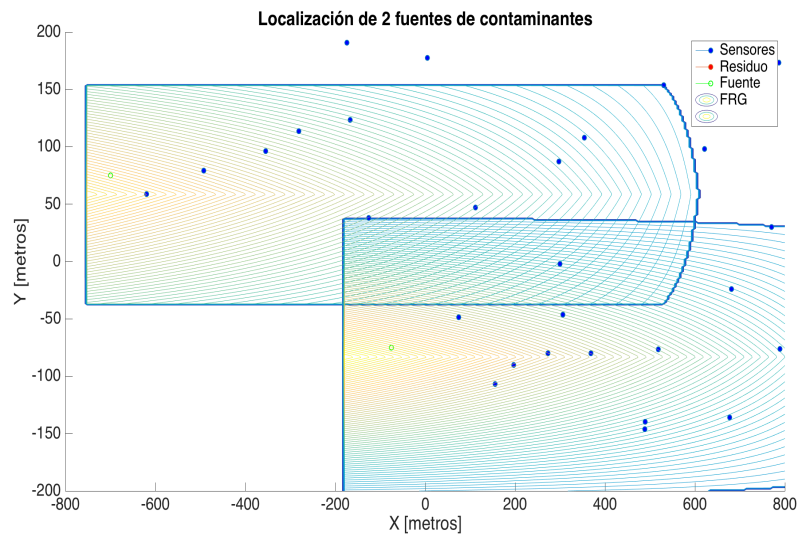


Ilustración 28: Localización de dos fuentes mediante ajuste de función FRG (III)

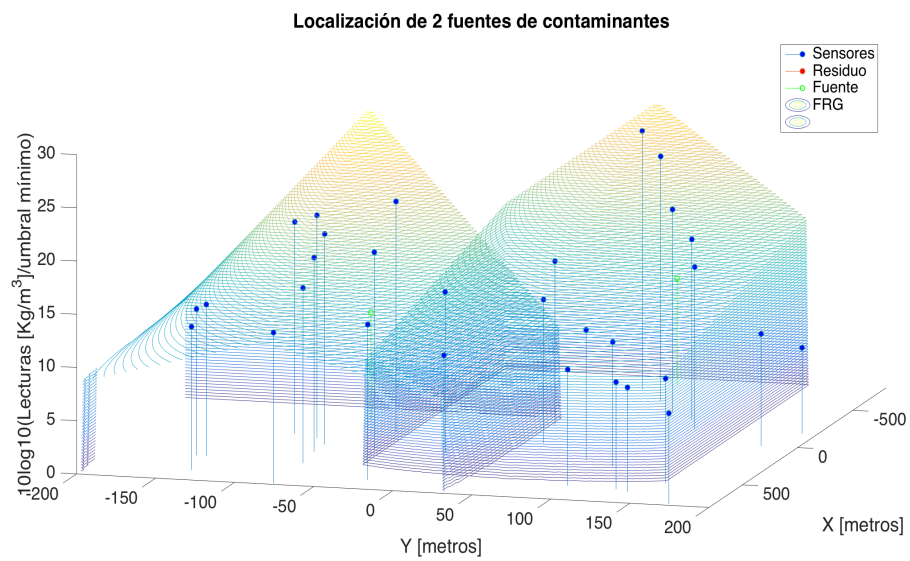


Ilustración 29: Localización de dos fuentes mediante ajuste de función FRG (IV)

5. Conclusions

As we have seen throughout this Degree Final Project, locating pollution sources is a difficult problem to deal with due to its complexity.

Firstly, using the Gaussian Plume model to locate sources involves knowing atmospheric conditions, wind speed and direction, source position and the pollutant emission rate, among others. Therefore, the obtained results remain limited to idealized scenarios where more parameters as usual are known.

According to Gradient Descent, Adaptive Gradient and Newton's Method, it can be observed a relation of dependence between the number of sensors and the necessary iterations to find the solution. Thus, the greater the number of sensors, the fewer iterations the algorithm will need. Furthermore, it has been identified the importance when choosing the initial point of such algorithms, since it may result in wrong solutions if chosen improperly.

Secondly, developing a locating method independent of any propagation model proves to be very useful, since we only need to know the sensor position and concentration measurements to locate the source, becoming this way a more realistic solution.

It was observed that the accuracy of localization was lower when using methods independent of a propagation model, but it is worth using it because the obtained results are so much closer to reality and therefore the number of parameters upon which the solution depends is lower.

At last but not least, the same method to locate two sources was successfully improved. Moreover, it only depended on sensor position and measurements. Nevertheless, in order to obtain the right results, sources had to be separated enough for anyone to distinguish two pollutant concentration maxima, each of which would be identified as a different source.

In my opinion, the obtained results have been positive and we have met the objectives proposed. However, there is still so much work to do in order to keep improving this project.

It is crucial to keep investigating how to control and reduce pollution, since it is a growing concern and an increasingly harmful problem for the health of living beings.

6. Líneas futuras

A lo largo del proyecto se identificaron algunas líneas posibles de mejora que serán expuestas a continuación:

- Realizar pruebas de verificación en escenarios reales.
- Implementación de un algoritmo mejorado para localización de fuentes en escenarios más realistas (evitar la simplificación de “tierra plana”; situar obstáculos entre fuente y sensores; probar escenarios de gran tamaño tipo comarcas, localidades o ciudades).
- Comparar el modelo de propagación utilizado para simular las medidas de concentración de contaminación con otros especificados en [6].
- Plantear el problema de monitorización en tiempo real de los sensores para intentar un procesamiento continuo de las medidas [14] y que la localización de fuentes de contaminantes pueda realizarse en un escenario más real en el que todos los parámetros atmosféricos (tales como dirección y velocidad del viento, temperatura, etcétera.) y de entorno varían en tiempo real.
- Implementar una función en el algoritmo que calcule la calidad de la estimación, que determine un porcentaje de probabilidad de que la fuente de contaminantes se encuentra en la posición estimada. Puede ser útil la referencia [22].
- Probar el algoritmo en escenarios como ciudades teniendo en cuenta la topología del problema y la posible necesidad de un modelo de propagación micro climático (para entornos urbanos), para la localización de fuentes de contaminante en ciudades [7].



7. PRESUPUESTO DE PROYECTO

1.- Autor: Francisco David Reyes Guerrón

2.- Departamento: Teoría de la Señal y Comunicación

3.- Descripción del Proyecto:

- Título: Localización de fuentes de emisiones de contaminantes mediante redes de sensores
- Duración (meses): 9
Tasa de costes Indirectos: 20%

4.- Presupuesto total del Proyecto (valores en Euros):

17.989 Euros

5.- Desglose presupuestario (costes directos)

PERSONAL

Apellidos y nombre	Categoría	Dedicación (horas totales)	Coste hora (euro a)	Coste (Euro)	Firma de conformidad
González Serrano, Francisco Javier	Ingeniero	90	18,00	1.620,00	
Reyes Guerrón, Francisco David	Ingeniero Junior	520	13,00	6.760,00	
Total				8.380,00	

a) Valor medio

Máximo anual para PDI de la Universidad Carlos III de Madrid de 1.155 horas

EQUIPOS

Descripción	Coste (Euro)	% Uso dedicado proyecto	Dedicación (meses)	Periodo de depreciación	Coste imputable d)
Ordenador	1200,00	100	9	60	180,00
Total					180,00

d) Fórmula de cálculo de la Amortización:

$$\frac{A}{B} \times C \times D$$

A = nº de meses desde la fecha de facturación en que el equipo es utilizado

B = periodo de depreciación (60 meses)

C = coste del equipo (sin IVA)

D = % del uso que se dedica al proyecto (habitualmente 100%)

SUBCONTRATACIÓN DE TAREAS

Descripción	Empresa	Coste imputable
	Total	0,00

OTROS COSTES DIRECTOS DEL PROYECTO)

Descripción	Empresa	Costes imputable
Licencia Microsoft Office		149,00
Matlab 2016b + Optimization toolbox		3.150,00
Material de Oficina		50,00
Internet		480,00
	Total	3.829,00

6.- Resumen de costes

Presupuesto Costes Totales	Presupuesto Costes Totales
Personal	8.380,00
Amortización	180,00
Subcontratación de tareas	0
Costes de funcionamiento	3.829,00
Costes Indirectos	2.477,80
Total sin IVA	14.866,80
Total con IVA	17.988,83

“El presupuesto total de este proyecto asciende a la cantidad de 17.989 EUROS”.

Leganés a X de Junio de 2016

El ingeniero proyectista

Fdo. Francisco David Reyes Guerrón

8. Bibliografía

- [1] Arystanbekova NK. Application of Gaussian plume models for air pollution simulation at instantaneous emissions. *Math Comput Simul* 2004;67(4):451-458.
- [2] Busquet E, Casas A, Pinto V, Rivero L, Sabadía J. Geofísica ambiental: técnicas no destructivas para el reconocimiento de zonas contaminadas por vertidos. *Acta geológica hispánica* 1995;30(1):73-82.
- [3] Chang PP, Yang JR, Den W, Wu C. Characterizing and locating air pollution sources in a complex industrial district using optical remote sensing technology and multivariate statistical modeling. *Environmental Science and Pollution Research* 2014;21(18):10852-10866.
- [4] Duchi J, Hazan E, Singer Y. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research* 2011;12:2121-2159.
- [5] Empresa Dräger. Catálogo de Detección de gases de Dräger. 2009. <http://www.draeger.com/sites/assets/PublishingImages/Segments/CSA/Oil-and-Gas-Industry/Download-Center/Portable-Gas-Detection-Devices/Detecci%F3n%20de%20gases.pdf>¹
- [6] European Environment Agency. The application of models under the European Union's Air Quality Directive: A technical reference guide 2011.
- [7] Fernández Fernández FJ. Simulación y caracterización de un incidente NBQ de tipo químico en un escenario urbano 2015.
- [8] Fletcher R, Powell MJ. A rapidly convergent descent method for minimization. *The Computer Journal* 1963;6(2):163-168.
- [9] Henry RC. Locating and quantifying the impact of local sources of air pollution. *Atmos Environ* 2008;42(2):358-363.
- [10] Igual García J. Algoritmos iterativos: método del gradiente descendente para calcular el mínimo de una función. 2011.
- [11] Islam M, Roy G. A mathematical model in locating an unknown emission source. *Water Air Soil Pollut* 2002;136(1-4):331-345.
- [12] Khedo KK, Perseedoss R, Mungur A. A wireless sensor network air pollution monitoring system. *arXiv preprint arXiv:1005.1737* 2010.

¹ Revisada a fecha Mayo de 2016

- [13] Lin C, Wu Y, Chang K, Lai C. A method for locating influential pollution sources and estimating their contributions. *Environmental Modeling & Assessment* 2004;9(2):129-136.
- [14] Ma Y, Richards M, Ghanem M, Guo Y, Hassard J. Air pollution monitoring and mining based on sensor grid in London. *Sensors* 2008;8(6):3601-3623.
- [15] Park SK, O'Neill MS, Stunder BJ, Vokonas PS, Sparrow D, Koutrakis P, et al. Source location of air pollution and cardiac autonomic function: trajectory cluster analysis for exposure assessment. *Journal of Exposure Science and Environmental Epidemiology* 2007;17(5):488-497.
- [16] Raju PV, Aravind R, Kumar BS. Pollution Monitoring System using Wireless Sensor Network in Visakhapatnam. *International Journal of Engineering Trends and Technology* 2013;4(4):591-595.
- [17] Shanno DF. Conditioning of quasi-Newton methods for function minimization. *Mathematics of computation* 1970;24(111):647-656.
- [18] Sohn MD, Reynolds P, Singh N, Gadgil AJ. Rapidly locating and characterizing pollutant releases in buildings. *J Air Waste Manage Assoc* 2002;52(12):1422-1432.
- [19] Spinelle L, Gerboles M, Villani MG, Aleixandre M, Bonavitacola F. Field calibration of a cluster of low-cost available sensors for air quality monitoring. Part A: Ozone and nitrogen dioxide. *Sensors Actuators B: Chem* 2015;215:249-257.
- [20] Stockie JM. The mathematics of atmospheric dispersion modeling. *SIAM Rev* 2011;53(2):349-372.
- [21] Tsujita W, Yoshino A, Ishida H, Moriizumi T. Gas sensor network for air-pollution monitoring. *Sensors Actuators B: Chem* 2005;110(2):304-311.
- [22] Uria-Tellaetxe I, Carslaw DC. Conditional bivariate probability function for source identification. *Environmental Modelling & Software* 2014;59:1-9.
- [23] Valdés DR, Pérez LE, Santos OAC, Aranda AEC, Díaz AS, Díaz ALG. Métodos teóricos para calcular parámetros tecnológicos de emisión de contaminantes atmosféricos producidos por fuentes fijas. Caso de estudio ciudad de Pinar del Río. *Nereis: revista iberoamericana interdisciplinar de métodos, modelización y simulación* 2012(4):19-30.
- [24] Vukovic V, Tabares-Velasco PC, Srebric J. Real-time identification of indoor pollutant source positions based on neural network locator of contaminant sources and optimized sensor networks. *J Air Waste Manage Assoc* 2010;60(9):1034-1048. ²
- [25] <http://www.elmundo.es/salud/2016/01/18/569bba3d268e3ea1548b45e4.html>

² Revisada a fecha Mayo de 2016

[26] <http://www.quiminet.com/articulos/la-importancia-del-zinc-en-la-industria-2585268.htm>³

[27] Zhang, X., Yang, L., Li, Y., Li, H., Wang, W., & Ye, B. (2012). Impacts of lead/zinc mining and smelting on the environment and human health in China. *Environmental monitoring and assessment*, 184(4), 2261-2273.

[28] <https://courses.ecampus.oregonstate.edu/ne581/eleven/plume.htm>

[29] <http://www.onmyphd.com/>⁴

³ Revisada a fecha Mayo de 2016

⁴ Revisada a fecha Mayo de 2016

ANEXO 1: Fórmulas de interés

Derivadas parciales del modelo Gaussian Plume:

$$\begin{aligned} \frac{\partial f(x_s, y_s, z_s)}{\partial x_s} = & \left(\left(\frac{Q}{2\pi U a_y^{b_y} a_z^{b_z} |x - x_s|^{b_y+b_z}} \right) \left(-\frac{2b_y \left(\frac{-0.5(y-y_s)^2}{a_y^2} \right) \left(e^{\left(\frac{-0.5(y-y_s)^2}{a_y^2 |x-x_s|^{2b_y}} \right)} \right) \left(\frac{x_s - x}{|x - x_s|} \right)}{|x - x_s|^{2b_y+1}} \right) \right. \\ & + \left(-\frac{Q(b_y + b_z)}{2\pi U a_y^{b_y} a_z^{b_z} |x - x_s|^{b_y+b_z+2}} \right) \left(e^{\left(\frac{-0.5(y-y_s)^2}{a_y^2 |x-x_s|^{2b_y}} \right)} \right) \left(e^{\left(\frac{-0.5(z-z_s)^2}{a_z^2 |x-x_s|^{2b_z}} \right)} + e^{\left(\frac{-0.5(z+z_s)^2}{a_z^2 |x-x_s|^{2b_z}} \right)} \right) \\ & + \left(\frac{Q}{2\pi U a_y^{b_y} a_z^{b_z} |x - x_s|^{b_y+b_z}} \right) \left(e^{\left(\frac{-0.5(y-y_s)^2}{a_y^2 |x-x_s|^{2b_y}} \right)} \right) \left(\left(-\frac{2b_z \left(\frac{-0.5(z-z_s)^2}{a_z^2} \right) e^{\left(\frac{-0.5(z-z_s)^2}{a_z^2 |x-x_s|^{2b_z}} \right)} \left(\frac{x_s - x}{|x - x_s|} \right)}{|x - x_s|^{2b_z+1}} \right) \right. \\ & \left. \left. + \left(-\frac{2b_z \left(\frac{-0.5(z+z_s)^2}{a_z^2} \right) e^{\left(\frac{-0.5(z+z_s)^2}{a_z^2 |x-x_s|^{2b_z}} \right)} \left(\frac{x_s - x}{|x - x_s|} \right)}{|x - x_s|^{2b_z+1}} \right) \right) \right) \right) \\ \\ \frac{\partial f(x_s, y_s, z_s)}{\partial y_s} = & \left(\frac{Q}{2\pi U a_y^{b_y} a_z^{b_z} |x - x_s|^{b_y+b_z}} \right) \left(\frac{(y - y_s)}{a_y^2 |x - x_s|^{2b_y}} \right) \left(e^{\left(\frac{-0.5(y-y_s)^2}{a_y^2 |x-x_s|^{2b_y}} \right)} \right) \left(e^{\left(\frac{-0.5(z-z_s)^2}{a_z^2 |x-x_s|^{2b_z}} \right)} + e^{\left(\frac{-0.5(z+z_s)^2}{a_z^2 |x-x_s|^{2b_z}} \right)} \right) \\ \\ \frac{\partial f(x_s, y_s, z_s)}{\partial z_s} = & \left(\frac{Q}{2\pi U a_y^{b_y} a_z^{b_z} |x - x_s|^{b_y+b_z}} \right) \left(e^{\left(\frac{-0.5(y-y_s)^2}{a_y^2 |x-x_s|^{2b_y}} \right)} \right) \left(\left(\frac{(z - z_s) e^{\left(\frac{-0.5(z-z_s)^2}{a_z^2 |x-x_s|^{2b_z}} \right)}}{a_z^2 |x - x_s|^{2b_z}} \right) + \left(-\frac{(z + z_s) e^{\left(\frac{-0.5(z+z_s)^2}{a_z^2 |x-x_s|^{2b_z}} \right)}}{a_z^2 |x - x_s|^{2b_z}} \right) \right) \end{aligned}$$

ABSTRACT

Introduction

Nowadays, particular attention is being paid to the release of pollutants through several pollution sources. Day after day, the health of any living being is harmed because of the pollution generated by different industrial activities or the burning of fossil fuels, among others.

According to the UN, “pollution in cities contributes to almost 3.4 million of premature deaths around the world and it is a key factor in respiratory and cardiovascular diseases”, which is a matter of concern.

This Degree Final Project is aimed at identifying the location of these pollution sources in order to carry out a more efficient control over air pollution.

To undertake this task, we must have a sensor network that allows collecting measurements of pollutant concentration in a given context, as well as other interesting parameters, such as wind speed and direction, to process them with a set of iterative algorithms which facilitate the localization of the source.

Lacking a real scenario to perform the relevant tests, it was decided to choose a full simulation by means of the so-called Matlab development environment and the aforementioned Gaussian Plume model. Thus, we can obtain the simulated measurements of pollutants which sensors would collect in a real scenario.

The algorithm will use these measurements to reproduce the pollutant propagation model and therewith locate the source.

Gaussian Plume Model

The Gaussian Plume Model is a purely mathematical model that allows representing air pollution propagation in a simplified manner. It is based on the convection-diffusion equation.

Its purpose is to represent in a single compact equation all the possible chemical-physical phenomena such as wind speed, environment temperature, pressure or chemical reactions between pollutants and the environment.

The hypotheses upon which the Gaussian Plume model is based are the following:

- Model development is implemented for a steady state
- Mass diffusion is despicable in the X axis
- Wind speed is constant; wind speed variations are deemed to be despicable
- The pollution source has a constant and continuous emission rate
- Flat Earth hypothesis

The main disadvantage regarding the use of the Gaussian Plume model is the necessity of previous hypotheses, since sometimes these become limitations to the study of the pollutant propagation (e.g. a constant and unidirectional wind, a constant pollutant emission rate, etc.). However, these are necessary to simplify and implement the model. Despite its limitations, the Gaussian Plume model is very useful, because it allows an acceptable representation of air pollutant dispersion. Besides, given the great number of parameters involved, we can perform all kinds of simulations.

Proposed solutions

Two different solutions are proposed to tackle the problem of source localization. Both share the same idea of minimizing the mean squared error (MSE) or cost function between real and approximate readings (obtained by means of a set of optimisation algorithms).

The first solution was to implement iterative optimisation methods such as the Gradient Descent, the Adaptive Gradient and the Newton's Method and attempt to minimize a cost function, which in this case depends on the Gaussian Plume model.

The Gradient Descent is a simple iterative algorithm which mainly depends on the gradient of the function to be minimized and a constant known as learning rate. The value which configures the learning rate is essential, since, if defined improperly, the algorithm may result in wrong solutions.

The Gradient Adaptive was the second method implemented. It also depends on the Gradient of the function to be minimized and, in this case, it is used an adaptive learning rate rather than a constant one. This allows us to obtain a solution with fewer iterations in comparison with the Gradient Descent method.

The Newton's Method was the third and last method implemented in this part of the solution. This makes use of the gradient and the Hessian matrix of the cost function to be minimized in order to find a solution to the optimisation problem. Despite being more complex than the other two algorithms, it requires fewer iterations to find the solution.

In the aforementioned methods, it is intended to estimate the real sensor measurements by searching the source position. When the estimated position of the source and the real one match, the MSE between the readings will be zero.

The cost function derives from the Gaussian Plume model, since, this way, algorithms tend to find the optimal position of the source minimizing the error among the measurements.

As for the second solution, it was suggested to find a general way to locate pollution sources avoiding the use of atmospheric propagation models that depend on a vast number of parameters and hypotheses, such as the Gaussian Plume model.

The purpose was to find a source-locator algorithm that only depended on sensor positions and pollution measurements.

Firstly, it was considered to implement a mathematical function similar to the Gaussian Plume model that depended on fewer parameters, but due to its complexity, it became unfeasible. Then it was found out that, if working with a logarithmic scale of measurements, the problem got easier and an almost linear variation among sensor measurements was observed.

Thus, it was planned to implement a function whose partial derivatives in the X and Y axes were linear. Besides, this function should depend on a few parameters. This function was called "FRG function" and is described below:

$$FRG = \frac{h}{h + \epsilon} \left(\frac{x_c - x}{x_{max}} + 1 \right) \left(-\frac{|y - y_c|}{y_{max} + \epsilon} + 1 \right) + h_0$$

Where:

h : Function relative height.

h_0 : Function base height.

x : X-coordinate in which the FRG function is evaluated.

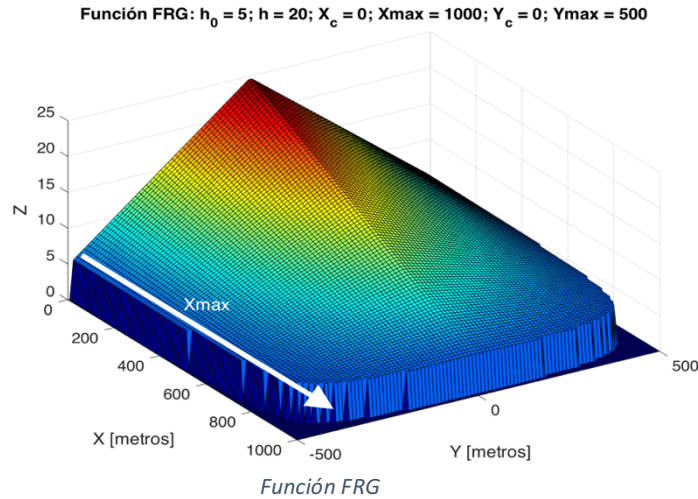
x_c : X-coordinate in the centre of the FRG function.

x_{max} : Maximum range of the FRG function in the X-axis.

y : Y-coordinate in which the FRG function is evaluated.

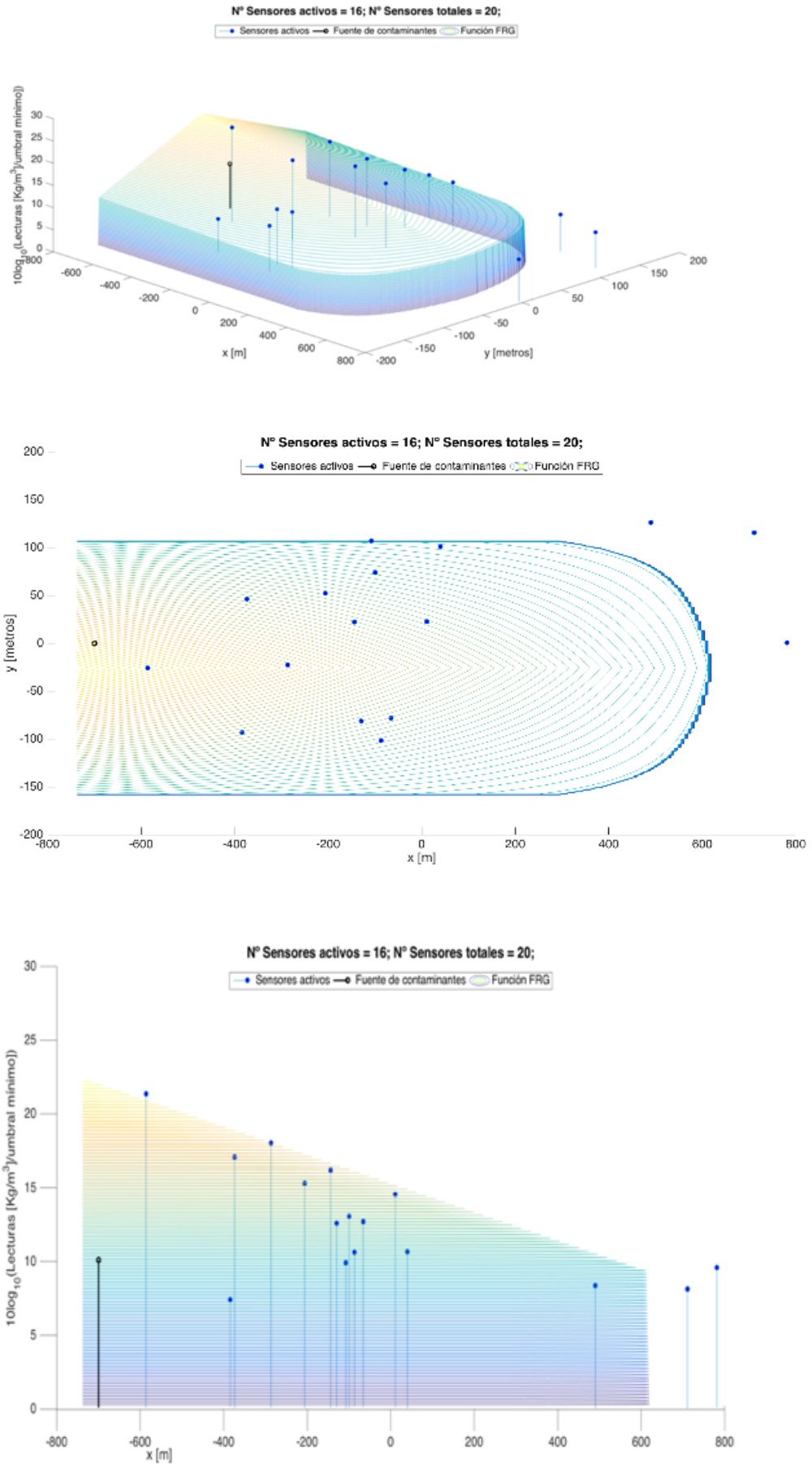
y_c : Y-coordinate in the centre of the FRG function.

y_{max} : Maximum range of the FRG function in the Y-axis.



Now, the purpose of this function would be to fit its parameters to get as close as possible to the measurements which sensors were providing, since the function height represents the pollutant concentration (in a logarithmic scale). Therefore, we used the Matlab function “lsqcurvefit”, which allows fitting the input function parameters in an optimal way by using a set of observations.

The obtained results were positive due to the flexibility of the FRG function and because the logarithmic scale of measurement contributed to a linear variation of the sensor real measurements. Some results derived from the implementation of this method to locate a pollution source are shown below:

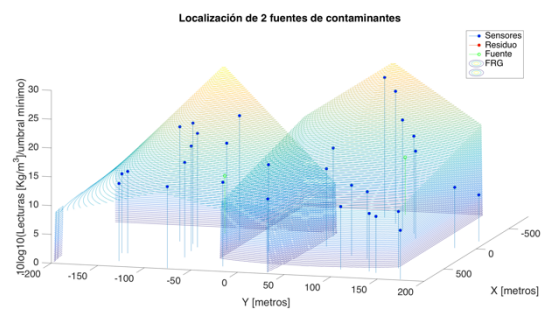
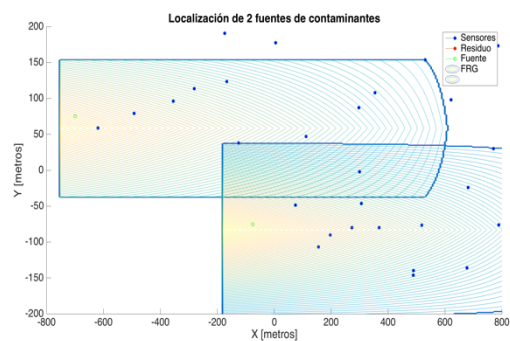
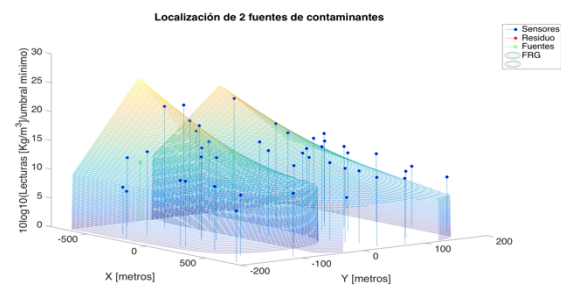
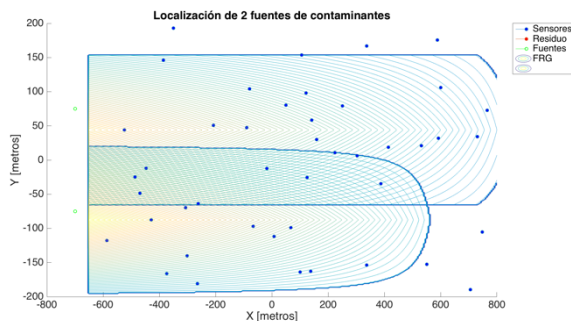


At this point, we analysed if there was any possibility to locate two pollution sources by using the same method. As a conclusion, it was necessary that both sources were separated enough from each other, so that the method could detect two maxima (corresponding to both pollution sources) and locate their positions.

To implement this improvement in the method, we considered to perform the first algorithm iteration after introducing the sensor measurements as observations in the “lsqcurvefit” function. Then, a second iteration in which we would introduce, as an observation, the residue resulted from the first iteration estimated measurements and the sensor real measurements. This way, all the measurements are justified in both iterations.

As it has been assumed that both sources are separate enough from each other, the algorithm cannot justify all observations in the first iteration, leaving a pollutant concentration maximum to be estimated in the second iteration.

Thus, in the first iteration the algorithm locates a first source and, in the second one, the other source. Some results of the algorithm to locate two pollution sources for two different scenarios are shown below:



Conclusions

As we have seen throughout this Degree Final Project, locating pollution sources is a difficult problem to deal with due to its complexity.

Firstly, using the Gaussian Plume model to locate sources involves knowing atmospheric conditions, wind speed and direction, source position and the pollutant emission rate, among others. Therefore, the obtained results remain limited to idealized scenarios where more parameters as usual are known.

According to Gradient Descent, Adaptive Gradient and Newton's Method, it can be observed a relation of dependence between the number of sensors and the necessary iterations to find the solution. Thus, the greater the number of sensors, the fewer iterations the algorithm will need. Furthermore, it has been identified the importance when choosing the initial point of such algorithms, since it may result in wrong solutions if chosen improperly.

Secondly, developing a locating method independent of any propagation model proves to be very useful, since we only need to know the sensor position and concentration measurements to locate the source, becoming this way a more realistic solution.

It was observed that the accuracy of localization was lower when using methods independent of a propagation model, but it is worth using it because the obtained results are so much closer to reality and therefore the number of parameters upon which the solution depends is lower.

At last but not least, the same method to locate two sources was successfully improved. Moreover, it only depended on sensor position and measurements. Nevertheless, in order to obtain the right results, sources had to be separated enough for anyone to distinguish two pollutant concentration maxima, each of which would be identified as a different source.

In my opinion, the obtained results have been positive and we have met the objectives proposed. However, there is still so much work to do in order to keep improving this project.

It is crucial to keep investigating how to control and reduce pollution, since it is a growing concern and an increasingly harmful problem for the health of living beings.

Future developments

Throughout this project, we have identified some possible future developments which are described below:

- To perform verification tests in real scenarios.
- To implement an improved algorithm to locate sources in more realistic scenarios by avoiding the Flat Earth simplification, placing obstacles between the source and the sensors and trying big-size scenarios such as towns, cities or regions.
- To compare the propagation model used to simulate the pollution concentration measurements with others specified in [6].
- To pose the problem of a real-time monitoring of the sensors to get a continuous processing of measurements [14] and also a more real scenario to perform the localization of pollution sources in which all the atmospheric and environmental parameters (such as wind speed and direction, temperature, etc.) vary in real time.
- To implement a function in the algorithm that calculates the quality of the estimation by determining a probability percentage that the pollution source is in the estimated position. It may be useful reference [22].
- To test the algorithm in scenarios such as cities, bearing in mind the type of problem and the possible necessity of a microclimatic propagation model (designed to urban environments) to locate pollution sources in cities.